

Generating Explanatory Discourse: A Plan-Based, Interactive Approach

Alison Cawsey

Ph.D.

University of Edinburgh
1989



**I DECLARE THAT THIS THESIS HAS BEEN COMPOSED
BY MYSELF AND THAT THE WORK DESCRIBED IN IT
IS MY OWN:**

(Alison Cawsey)

Acknowledgements

Thanks must first go to my supervisors, Peter Ross and Paul Brna, for helpful criticism and encouragement throughout. The people who have taken part in my various experiments or provided useful comments on my work are too numerous to mention, but include Judy Delin, Karen Valley, Diana Bental, Jean Carletta, Kathleen King, Mitch Harris, Maggie Cooper and Colin Williams. Thanks especially to Chris Mellish, for making the time to make helpful comments on my papers and this thesis; and to Richard Tobin for encouragement and useful comments on my work, putting in the missing hyphens and apostrophes, fixing computers, helping me write better code, making me nice food, and being wonderful. Oh yes, and Alan Black for pizzas and chocolate ice-cream and for testing the cookies.

I should also acknowledge the Science and Engineering Research Council, for financial support.

Published Papers

The following published papers report aspects of the work described in this thesis:

- Some of the initial analysis of explanations, described in chapter 3, is published as “Explaining the Behaviour of Simple Electronic Circuits” in the proceedings of the *International Conference of Intelligent Tutoring Systems* (1988).
- The basic discourse planner, described in chapters 4-6, is briefly described in “The Structure of Tutorial Discourse”, published in the proceedings of the *4th International Conference on AI and Education* (1989).
- A description of how explanations are generated is given in “Explanatory Dialogues”, published in the journal *Interacting with Computers*. (1989)

Abstract

This thesis presents a computer model of 'explanatory discourse'. As computers are increasingly used to present advice, help or explanations to novice users, it is important to consider how best to structure the resulting discourse. This involves organising what to say (depending on the user's knowledge and the structure of the domain knowledge), and managing any dialogue (such as interruptions). In a tutorial context it may also involve responding to perceived misconceptions, involving the novice in the explanation and checking their understanding. As the explanatory dialogue progresses, assumptions about the user's knowledge must be updated.

No previous system incorporates all these aspects in an integrated model. Yet each aspect of the system depends on the others. The structure of the text is related to the conceptual structure of the domain (and hence to the model of the user's knowledge) while the goal of inferring what the user knows may influence the dialogue.

The model described in this thesis is based on analyses of human face to face explanations, and on related work on tutorial dialogues, discourse analysis and text generation. The main examples used throughout are tutorial explanations of how simple circuits work. However, the framework developed generalises both to other domains and to other types of discourse. The model is fully implemented and an initial evaluation has been done, suggesting that it has potential as a practical approach to generating explanations in tutorial, help or advice systems. It therefore contributes both to our understanding of human explanatory discourse, and to our understanding of how to use computers more effectively to give complex explanations.

Table of Contents

1. Introduction	1
1.1 Motivation: Advice, Explanation and Dialogue	3
1.1.1 Negotiation in Help and Advice	3
1.1.2 Checks and Repair	4
1.1.3 Follow-Up Questions	4
1.1.4 Getting the Level Right: Meta-dialogues	5
1.1.5 Types of Interaction	5
1.1.6 A User Model?	6
1.2 Approach: Planning and Reacting	6
1.3 Criteria: Features of Explanatory Discourse	9
1.4 An Outline of the Thesis	10
2. Review: Text and Dialogue Structure	13
2.1 Organising the Information	14
2.1.1 Generating Expository Text	14
2.1.2 Curriculum Planning	21
2.1.3 Conclusion: Organising Content	26
2.2 Structuring Dialogue	26

2.2.1	Tutorial Dialogues	26
2.2.2	Linguistic Theories of Discourse Structure	31
2.3	Text and Discourse Structure in Explanatory Dialogues	42
2.3.1	Structuring the Explanation Content	43
2.3.2	Structuring Interactions	43
2.3.3	Inferring and Using a Model of the User's Knowledge	45
2.4	Conclusion	46
3.	Analysing Human Explanations	48
3.1	Types of Explanation	49
3.2	Text Structure: Schemas, Grammar Rules and Plans	53
3.2.1	Classifying Utterances	54
3.2.2	Describing Structure: Schemas	56
3.2.3	Describing Structure: Grammar Rules	59
3.2.4	Explaining Structure: Plans and Discourse Models	60
3.2.5	Conclusion	63
3.3	Dialogue Structure	64
3.3.1	Dialogue as Interrupted Text?	64
3.3.2	Exchanges and Interaction	70
3.3.3	Topic Structure and Focus	85
3.3.4	Reacting to Misconceptions	87
3.4	Making Choices (and other issues)	89
3.5	Conclusion	90

4. Structuring Communicative Text	92
4.1 Criteria for Generating Communicative Text	92
4.1.1 Standards of Textuality	93
4.1.2 Domain Dependent Criteria	104
4.1.3 Summary	105
4.2 The EDGE Text Planning System	106
4.2.1 The EDGE Discourse Planner	108
4.2.2 Planning Text Content	113
4.2.3 Planning the Presentation	115
4.2.4 Varying the Domain or Situation	121
4.2.5 Summary: Planning Communicative Text	125
4.3 Conclusion	126
5. Controlling Interactions	128
5.1 Criteria for Structuring Discourse	128
5.1.1 Principles from Linguistic and Sociolinguistic Studies . .	129
5.1.2 Principles from Work on Tutorial Dialogues	133
5.1.3 Summary	134
5.2 The EDGE Discourse Planner	135
5.2.1 The Basic Model: Structuring Sequences of Exchanges . .	136
5.2.2 Extensions: Interruptions and Remediation	144
5.2.3 Limitations of the Model	152
5.2.4 Example Interactions	153

5.2.5	Generating Other Discourse Types	156
5.3	Conclusion: Local Interactions in a Planned Explanation	159
6.	Updating the User Model	163
6.1	Review: Inferring the User Model	163
6.1.1	Stereotypes	164
6.1.2	Inference Rules and Conceptual Structure	165
6.1.3	Observing the User	167
6.1.4	Degrees of Belief	168
6.1.5	Summary: Sources of Information	168
6.2	Criteria for a User Modelling Component	169
6.2.1	Indirect Inference	169
6.2.2	Dealing with Contradictions and Changing Assumptions .	170
6.2.3	Using and Inferring User Model in Discourse	171
6.2.4	Summary	171
6.3	The EDGE User Modelling System	172
6.3.1	Inferring the User Model	173
6.3.2	Effect on Discourse	180
6.3.3	Example	181
6.3.4	Other Types of Discourse	182
6.4	Conclusion	185
7.	Application and Examples: How Circuits Work	187
7.1	Review: Explaining How Things Work	188

7.2	Criteria for Explaining How Things Work	191
7.3	The EDGE Circuit Explainer	193
7.3.1	Plans to Explain Circuits	193
7.3.2	Device Models, Instances and Diagrams	202
7.3.3	Example Explanations	203
7.3.4	Explaining Other Things	210
7.4	Conclusion	215
8.	Evaluation, Assessment and Further Work	217
8.1	Empirical Evaluation	218
8.1.1	Type of Evaluation	219
8.1.2	Identifying Problems	220
8.1.3	Examining the Scripts	225
8.1.4	The User's Opinion	225
8.1.5	Conclusion	227
8.2	Assessment and Comparison	227
8.2.1	Limitations in the Planning Formalism	228
8.2.2	Local Responses in a Changing Situation	231
8.2.3	Limitations of the User Model	234
8.2.4	Summary	235
8.3	Further Work	235
8.3.1	Explanation Content	236
8.3.2	Interactions and Dialogue	237

8.3.3	The User Model	239
8.3.4	Applications	239
8.4	Conclusion	240
9.	Conclusion	242
9.1	Summary — The Story So Far	242
9.2	Where Has it Got Us?	244
9.2.1	The System: Informative, Interactive, Individualised Explanations	244
9.2.2	The Approach: Integrating Ideas	245
9.2.3	The Key Ideas	246
9.3	Where Now?	247
	References	248
A.	Human Explanations of Circuits	266
A.1	Uninterrupted Explanations	266
A.2	Text Book Explanations	269
A.3	Dialogues	271
A.3.1	Followup Questions	271
A.3.2	Interrupted Explanations	272
A.3.3	Guided Explanations	275
A.3.4	Explanations Dominated by the Novice	276
A.3.5	Complex Cooperative Explanation	277
A.4	Circuit Diagrams	280

B. Content Plans	283
C. Discourse Plans	290
C.1 Transactions	290
C.2 Exchanges	291
C.3 Moves	293
C.4 Acts	296
D. Remediation Plans	297
E. Device Models	299
E.1 Complete Circuits	299
E.2 Components and Circuit Types	304
F. The Evaluation Questionnaire	306
G. Example Dialogues with Real Users	308
G.1 User P: Managing Mixed Initiative Interaction	308
G.2 User M: Followup Questions and Remediation	309
G.3 User D: Managing Interruptions	311
G.4 User J: Repeated Explanations	312
H. Example Displays	315

Chapter 1

Introduction

This thesis presents a computer model of ‘explanatory discourse’. As computers are increasingly used to present advice, help or explanations to novice users, it is important to consider how best to structure the resulting discourse. This involves organising what to say (depending on the user’s knowledge and the structure of the domain knowledge), and managing any dialogue (such as interruptions). In a tutorial context it may also involve responding to perceived misconceptions, involving the novice in the explanation and checking their understanding. As the explanatory dialogue progresses, assumptions about the user’s knowledge must be updated.

No previous system incorporates all these aspects in an integrated model. Yet each aspect of the system depends on the others. The structure of the text is related to the conceptual structure of the domain (and hence to the model of the user’s knowledge) while the goal of inferring what the user knows may influence the dialogue.

The model described in this thesis¹ is based on analyses of human face-to-face explanations, and on related work on tutorial dialogues, discourse analysis and

¹Referred to from now on as the EDGE system (Explanatory Discourse GEnerator).

text generation. The main examples used throughout are tutorial explanations of how simple circuits work. However, the framework developed generalises both to other domains and to other types of discourse. The model is fully implemented and an initial evaluation has been done.

This thesis takes a broad view of explanation, not limited to justifications (answers to ‘why’ questions). This is consistent with Draper’s definition of explain as ‘make known in detail’, based on differences between the explainer’s and inquirer’s knowledge [Draper 87]. An explanation, then, consists of the responses to the following sorts of requests²:

- Explain who the queen is.
- Explain where you went.
- Explain what a knight’s move in chess is.
- Explain the rules of the game.

In each case there is a coherent piece of knowledge to convey. The explanation is complete when the hearer is satisfied with the reply and understands that piece of knowledge. In some cases a brief phrase may be all that is needed. In others the explanation may involve a complex dialogue taking many minutes. In this case the explanation can be seen as a type of goal directed conversation, where the goal is to convey or understand a complex piece of knowledge.

The rest of this chapter will look at some further motivation for work on explanation and dialogue, suggest a basic approach and outline the structure of the thesis.

²Taken from [Draper 87].

1.1 Motivation: Advice, Explanation and Dialogue

There is considerable evidence that effective explanations involve dialogue. Without feedback the expert has no idea whether the explanation is at the right level, and without the opportunity to interrupt or ask follow-up questions there is little the novice can do if the explanation is not understood or if it fails to address the novice's real goals [Rymaszewski 87]. In order to motivate the thesis, this section will briefly review some of the work that has been done on human explanations and dialogue.

1.1.1 Negotiation in Help and Advice

Several people have noted the importance of negotiation in advice-giving. Pollack, for example, found that advice-giving dialogues are best regarded as a negotiation process, where both the problem to be solved and the solution are jointly worked out in an extended dialogue [Pollack *et al* 82]. Gilbert makes similar observations, but emphasises how local coherence constrains the organisation of the dialogue [Gilbert 87a]. As the novice continues to talk about the problem, the expert attempts to find things to say which are coherent with the last utterance, yet contribute to the overall goal of the conversation. Gilbert and Frohlich [Frohlich 88] go on to show how principles of conversational interaction developed in the field of conversation analysis may be applied to the design of an advice system. They emphasise the importance of allowing a wide variety of dialogue moves (such as questions, requests or statements) at any point in the dialogue, rather than enforcing fixed question and answer sequences.

1.1.2 Checks and Repair

Another feature of verbal explanations is the presence of *checking moves* and conversational repair. Ringle and Bruce discuss how verbal communication is full of such moves [Ringle & Bruce 81]. Misunderstanding is signalled through implicit cues such as lack of 'go ahead' signal, direct questions from the explainee and explicit checks by the expert ('did you understand that?'). Repair is managed by rephrasing or giving more background information, for example. Conversation is seen as consisting of several types of move: domain-independent management moves to control interaction; checks and repair moves to keep it running smoothly; and transactional moves to transmit the domain information.

1.1.3 Follow-Up Questions

Communicative explanations also frequently involve 'follow-up' questions. Moore observed that in interactions between teaching assistants and novice computer users the novices frequently didn't understand the expert's response and asked follow-up questions [Moore 89]. The expert would then have to define terms, re-explain or provide background information. This is slightly different from the checking moves described by Ringle and Bruce, emphasising the dialogue which may go on after an explanation is complete. It appears then that there are two places for such checks and interruptions — at the end of a coherent piece of explanation (discourse segment) or after the relevant utterance.

1.1.4 Getting the Level Right: Meta-dialogues

In much of the work on explanation, the purpose of an explanation is seen as being to transmit some piece of complex information from speaker to hearer³. A useful explanation is one which contributes something new, yet makes sense by being related to things already known by the inquirer [Draper 87]. An explanation might therefore consist of just one new piece of information, and a whole lot of ‘anchoring’ information, linking in with existing knowledge. Planning an explanation is seen then as first deciding what new information to include, and then deciding what additional background information is necessary. Draper emphasises that because the expert cannot see into the novice’s mind, some sort of dialogue is essential if this process is to proceed smoothly. The interaction may involve negotiating what explanation was wanted [Gilbert 87b], as well as how much background information is needed. The explanation may therefore begin with a ‘meta-dialogue’ to find out this information.

1.1.5 Types of Interaction

The importance of interaction in explanation should be clear from this brief review. There are numerous types of interaction which may contribute to a successful explanation, including:

- Negotiation or meta-dialogues at the beginning to establish what explanation is sought and how much the novice knows.
- Checking moves and opportunities for interruption within the explanation.
- Follow-up questions at the end of the explanation.

³But see [O’Malley 87] for a different view, emphasising the joint construction of an explanation through interaction.

Some types of discourse will have other types of dialogue move, such as explicit checks on the novice's understanding in a tutorial dialogue.

1.1.6 A User Model?

Many of the researchers cited above, in emphasising the importance of dialogue, have played down the importance of a 'user model'. Gilbert, for example, only considers dialogue structure in the design of an advice system, while Moore has a simple user model, but does not update it. This is perhaps a reasonable reaction to the overemphasis on user modelling as a means for getting better, more appropriate explanations, and the initial unrealistic expectations in this respect. However, this thesis suggests that updating assumptions about what the user knows is also important, and should be considered in conjunction with a model of dialogue and interaction. Furthermore the goal of updating this model will have an effect on the dialogue itself.

1.2 Approach: Planning and Reacting

It should be clear from the above discussion that in order to generate effective explanations it is necessary to consider both the dialogue structure and how explanation content should depend on the user's knowledge. But how are such 'explanatory dialogues' generated? There are two possible approaches to consider. In the first approach, dialogues are seen as resulting from the plans and goals of the participants. Hobbs and Evans, for example, discuss how a piece of informal, apparently unstructured conversation can be viewed as planned behaviour [Hobbs & Evans 79]. Power showed how this approach could be implemented to generate simple goal directed conversations [Power 74]. This approach also provides the basis for recent work on discourse understanding (e.g., [Allen & Perrault 83]) and discourse structure [Grosz & Sidner 85].

The second approach to be considered views dialogue as being influenced chiefly by the situation. Utterances are only locally planned, given the previous utterance, the current situation and constraints imposed by the rules and conventions of conversation. This is essentially the approach advocated by work on conversation analysis (e.g., [Suchman 87, Sacks *et al* 78]). It is also consistent with much of the work on generating tutorial dialogues (e.g., [Collins & Stevens 82, Woolf & McDonald 84]) where the teacher is reacting to the errors and misconceptions revealed by the student.

For most types of discourse, both approaches apply. In conversation, the dominant constraint may be local coherence, while in a formal lecture a high level plan may be more important. Yet any face-to-face interaction is reactive (considering the bored or confused faces of the audience in a lecture, for example), and the most informal conversations involve goals (though these may be social goals such as to impress, and impose few constraints). A model of explanatory discourse must consider both global and local coherence; both high level goals and how to react to unpredicted situations.

McDonald, in his work on language generation, suggests that generation is a process of decision making under constraints [McDonald 83]. He applies this idea to the generation of sentences given a semantic message to convey. However, the same applies to generating multi-sentence discourse. Given some communicative goal, there are certain constraints on how this may be achieved. These constraints include local coherence and conversational conventions. A model of discourse generation should consider how to plan an interaction to achieve a communicative goal given these constraints. While generating complex expository discourse requires some planning of global content, it must at the same time take into account the fact that the effects of this planning are uncertain, and acknowledge the importance of local interactions to deal with specific problems.

How then do people plan complex expository discourse? If the whole dialogue was planned out in advance this would be very inefficient. Because of

the uncertainty in human interaction it would be necessary either to create a complex conditional plan, with explicit ways of dealing with every contingency, or to be willing to frequently replan the whole discourse. It seems unlikely that people do either of these, or that this is the best way to approach robust communication in human-computer interaction (HCI)⁴. A more plausible model is suggested by Young and Simon [Young & Simon 87]. They argue that people use partial plans, with abstract goals only expanded into detailed plans when needed⁵. Planning is interleaved with execution, so later planning may take into account knowledge of the situation not deducible beforehand. As unexpected situations develop, the plan may be extended in different ways to deal with these.

This approach is similar in some ways to McDonald's model of sentence generation. It too can be made to run in approximately linear time⁶, with planning and execution interleaved. It can be viewed as a compromise between approaches on the one hand which plan things out in advance, before they begin an utterance (existing systems which generate multi-sentence text), and on the other hand dialogue systems which react to situations as they arise, and which may have no notion of a global plan. Of course, this approach only works when there are well defined methods for satisfying communicative goals — when the speaker is an expert in the domain, and in explaining it [Young & Simon 87] [Card *et al* 83, ch5]. When the explanation is more complex it may be necessary

⁴But see section 8.2.1 for a discussion of this.

⁵Note that this approach will only work reliably if subtasks are designed to work in all situations, so no backtracking is needed through the space of partial plans. This idea of using robust skeletal plans is analogous to the behaviour based approach in robotics [Malcolm & Smithers 88].

⁶By approximately linear time we mean here that doubling the length of the explanation should approximately double the total amount of processing.

to think hard about how to explain, working it out in some detail. However, in this thesis we are concerned with 'expert' performance at explanation, where the explainer has a set of standard ways of explaining different things depending on the hearer's knowledge ('skeletal plans'). The approach will therefore be to construct partial plans, and expand or extend them when needed depending on the situation.

1.3 Criteria: Features of Explanatory Discourse

The previous two sections have considered why dialogue is important in explanations (with reference to a number of studies of human interaction), and looked at some approaches to the generation of explanatory discourse. A model of explanatory discourse must capture the dialogue phenomena observed, while remaining computationally tractable. In particular then, the model should have the following properties:

Informative: It should be possible to plan an explanation which may be understood by the hearer! This involves assumptions about the hearer's prior knowledge, and knowledge of prerequisite relations between skills and facts in the domain.

Interactive: The explanation should be presented as part of a dialogue. This might involve negotiating the topic and what the user understands beforehand; presenting the explanation (allowing interruptions, queries and checking moves); and negotiating the close of the topic, allowing follow-up questions if necessary.

Individualised: As the dialogue progresses, assumptions about the user's knowledge should be revised. This should be both on the basis of direct evidence such as what the user says, and indirect evidence such as what

the user is told, or the difficulties of related concepts which are believed known. This information should be used to tailor the explanation to the user.

The architecture needed for generating this sort of discourse needs to integrate top down planning of the explanation with a more reactive approach, allowing interruptions, changing assumptions about the user, repairs and elaborations. A simple approach, consistent with human performance and running in linear time, is to use partial plans. These are expanded when needed, and may be extended as suggested by the situation. The EDGE model uses two main types of plans. Domain-content plans represent sub-skill and prerequisite relationships between concepts, and are used to plan what to say. Discourse plans represent typical dialogue structures applicable to a particular discourse genre, and are a form of 'meta' planning — they are plans to present domain-content plans. Discourse and domain plans are used jointly to plan an explanatory dialogue.

1.4 An Outline of the Thesis

This thesis shows how effective explanatory discourse may be generated which combines plan-based and 'interactive' approaches to communication. It shows how a tutorial explanation can be seen as a microcosm of the whole tutorial process, involving reasoning about teaching goals, the user's knowledge and the structure of interactions. The framework developed can be used to generate explanations in different domains and with different interactional styles, yet is consistent with linguistic models of discourse structure.

The model has been used to generate interactive explanations of circuit behaviour, combining textual and graphical information. This particular implementation is a second contribution of the thesis. Explanations of circuit behaviour are shown to consist of much more than a simple trace of causal be-

haviour, but to involve extra 'support' information. The generation of such explanations should depend on the knowledge of the user.

The rest of this thesis will develop these ideas further. The overall aim is to develop a computational model of human explanatory discourse and consider whether this has potential for improving computer explanation systems. The methodology has therefore been to analyse human explanations and examine related work; to develop a computational model based on that analysis; and to evaluate that model (explanation system) with a small number of real users. These three steps are reflected in the structure of the thesis (though the evaluation represents a relatively minor part).

So, chapters 2-3 provide the background to the thesis. First, chapter 2 reviews work on generating multi-sentence text, curriculum planning, discourse structure and tutorial dialogues. Each of these approaches will prove inadequate alone for generating explanatory discourse with the above features, but provide many ideas for the different parts of the system. Next, chapter 3 examines a number of examples of human explanatory dialogues. This serves two purposes. First, it shows how explanations in a particular domain are structured, and provides a basis for the main implementation of the model in the domain of electronic circuits. Second, it shows how an 'explanatory dialogue' may be structured — involving interruptions, checks and follow-up questions.

The EDGE model of explanatory discourse is developed in chapters 4-6. Chapter 4 presents an approach to generating communicative expository text given knowledge of what the user knows. The approach is based on content plans for structuring what to say, and discourse plans for selecting how to present the information. Chapter 5 extends this model to allow interactions with the user. This involves extending the discourse plans to structure exchanges with the user and to deal with interruptions. There are also special plans for dealing with errors and misconceptions as they are detected. Finally, in chapter 6, the model

is extended again so that the system may guess what the user knows, and update it as the dialogue progresses.

Chapter 7 shows how the EDGE model may be used to generate explanations in a particular domain. Relevant work on explaining physical systems is reviewed, and plans developed for explaining electrical circuits, based on this work and the analysis in chapter 3. A number of detailed examples of explanations generated by the system are given, showing how the explanatory dialogue is planned and the user model updated.

In chapter 8 both the particular implementation and the general approach are evaluated. This involves first presenting an initial formative evaluation of the system, then discussing the theoretical limitations of the architecture. Both these levels of evaluation point to a number of suggestions for further work.

Finally, chapter 9 will summarise the main contributions of the research, and directions for further research.

Chapter 2

Review: Text and Dialogue Structure

There are a large number of disciplines which may contribute to a model of explanatory discourse. Some of the more obvious ones are linguistics (especially theories of discourse structure), psychology (including work on understanding and misunderstanding text) and education (again, work on learning and misunderstanding). Other contributing disciplines include sociology (e.g., conversation analysis), philosophy (e.g., philosophy of explanation) and mathematics (proof structures). While this thesis attempts to combine a number of insights from different areas, it would clearly be intractable to review all the relevant work. There is already the danger that in considering contributions from a number of domains the thesis does full justice to none.

This chapter will review some of the more directly relevant work within artificial intelligence (AI) and discourse analysis. This includes work on text generation, curriculum planning, tutorial dialogues and theories of discourse structure. Related work in other fields will be reviewed in the appropriate chapters.

The chapter will be divided into two main sections. The first will be concerned with work on organising information which is to be conveyed, while the second will be concerned with the dialogue which may take place in conveying that information.

2.1 Organising the Information

People decide how to organise information at many levels. At the lowest level we decide how to structure short utterances so they are coherent and understandable. At a higher level we may decide how to structure a technical paper or a lecture. And at a higher level still we may decide how to organise a whole curriculum spanning many years of education.

These different levels are reflected in the two areas of AI research which have been most concerned with the organisation of information. The first (coming from work on natural language generation) has been concerned with organising the content of generally quite short (paragraph sized) sections of text. The second (coming from work on intelligent tutoring systems) has been concerned with planning information in a curriculum. Both of these are relatively new areas of research, and are described below.

2.1.1 Generating Expository Text

There has been increasing interest recently in methods for generating coherent multi-sentence text from some underlying representation of knowledge. Although early work tended to concentrate on sentence level generation ('how to say it') rather than planning 'what to say', the situation has since become more balanced, with a significant amount of work on selecting and structuring text content depending on the user's knowledge, attitudes or goals.

The summary below indicates the scope of this work — fuller reviews are given in [Moore & Swartout 88a], who emphasise the importance of dialogue, and in [McKeown & Swartout 88].

2.1.1.1 Following a Given Knowledge Structure

Early systems for generating multi-sentence text generally based the structure of the text on the structure of the domain knowledge or trace of reasoning. For example, Davey's PROTEUS system [Davey 74] generated coherent descriptions of a game of Tic-Tac-Toe. The generation process included heuristics for splitting the game plan into sentences, and for selecting anaphoric and conjoining expressions. However, the plan of what to say was simply the trace of moves in the game. This was adequate in this simple domain, where temporal sequencing dominated the ordering constraints.

Another system for describing a well-defined procedure is the Yale CAD help system explainer [Cullingford *et al* 82]. This system generates explanations from a representation of various procedures in a CAD system. Using various simple heuristics these may be summarised depending on whether they have been explained before.

Explanations given by expert systems similarly tend to use a fixed content structure, this time derived from the rule trace. While this may be modified in some way dependent on the user's prior knowledge [Scott *et al* 84, Wallis & Shortliffe 84] or transformed to make the output more readable [Weiner 80], the basic text structure is again given, being based directly on the underlying knowledge. In this case it is the causal relations in the domain which have most influence on the text plan.

If explanations are to be generated it is important that the underlying knowledge representation is appropriately structured, containing the necessary justifications. Clancey showed how the 'Mycin' expert system could be reconfigured to make the structural, strategic and support knowledge explicit, and how this could be used in explanation [Clancey 83b, Aikins 83]. Swartout went further in making the underlying justifications for rules explicit by using an automatic programmer to 'compile' an expert system from the underlying domain knowl-

edge and general diagnostic strategies [Swartout 83]. The derivation of the rules could then be used in giving explanations to the user.

2.1.1.2 Selecting and Ordering Knowledge

The above systems base the generated text strongly on the structure of the underlying knowledge. However, in many domains the underlying knowledge appears to have much less structure, with no temporal or causal links to impose standard orderings. Mann and Moore attempted to generate multi-paragraph text from an underlying semantic net based representation of domain knowledge [Mann & Moore 81]. This involved ordering fragments and suggesting paragraph breaks, filtering to 'throw out' what the user already knows, and aggregating fragments into sentences. Another system using similar techniques was developed by Kukich [Kukich 83]. Her system abstracted summaries of important features from a stock market report and expressed these in natural language.

However, there was no consistent theory of text structure implicit in either system (apart from simple clause combining rules). The emphasis was on the selection of information and on how to break it up into sentences. More recent work has been concerned more with the overall structure of the discourse, and the relations which may exist between sections of text.

McKeown examined numerous descriptions of physical objects and derived schemas which captured the common structure of these descriptions [McKeown 85]. These were expressed in terms of 'rhetorical predicates' which aimed to be domain independent descriptions of items in the text. Focus constraints further restricted choices in the generation, resulting in coherent text in the restricted domain. Work on using focussing to constrain text organisation has been developed further by McCoy and Cheng, who use *focus trees* to represent focus and a set of rules to determine how focus can change [McCoy & Cheng 88].

A more general approach is to consider the relations which may apply between sections of text in any domain (such as compare-contrast or motivation-

enablement) [Hobbs 77, Weiner 79, Mann 87, Mann & Thompson 87]. Mann and Thomson enumerate 23 such relations, which may be applied hierarchically to describe the structure of text. As a step towards explaining why these relations may be used, they associate with each relation the intentions of the speaker which might lead to it being used, and the resulting affect on the hearer. This framework (Rhetorical Structure Theory or RST) has recently been used in a number of plan-based text generation systems [Moore & Swartout 88b, Hovy 88a]. In Hovy's system the content of the description is selected before the planning or text-structuring process, while Moore uses the planning process to both select and structure the content. This latter approach seems to be more principled being based more on the communicative goal of the speaker.

Recently, Hovy and McCoy have attempted to combine the focus tree approach with the use of rhetorical relations [Hovy & McCoy 89]. This is a step towards using multiple constraints to influence text structure.

A very different approach to planning text is used in the KAMP system developed by Appelt [Appelt 82]. This uses a fairly sophisticated planner to plan a short section of text given a communicative goal. The planning involves the selection of referring expressions, and makes no distinction between 'what to say' and 'how to say it' — both are aspects of the planning process. The final plan may involve both text and physical actions such as pointing. Unfortunately Appelt's approach seems currently to be intractable for practical generation systems generating multi-paragraph explanations. The original version took several hours to generate a short section of text! This was partly because of the inefficient axiomatisation of possible world semantics to reason about beliefs, and partly because of the complex planning process taking into account all possible interactions between actions in the plans.

2.1.1.3 Tailoring Output to the User

Much of the current work on text planning concerns how to tailor descriptions to particular users with different goals, background knowledge and attitudes.

Paris, for example, extends McKeown's work to consider the different schemas which are used in describing objects to experts or to novices [Paris 87]. She shows (from an analysis of adult and junior encyclopaedias) that descriptions directed at experts are mostly structural (describing the structure of the device), while those directed at novices are causal explanations. Her generation system may combine different schemas depending on whether the user is an expert or a novice about the subparts of the system. This is clearly an extension of the simple 'don't include what the user already knows' approach to tailoring explanations, but it seems doubtful whether the expert-novice distinctions used are necessarily the most appropriate ones. Indeed, it seems that familiarity with the structure of a device is a prerequisite to understanding a causal explanation.

Of course, the user may not just have missing knowledge, but may have misconceptions which should be corrected. McCoy considers how to respond to 'object-related' misconceptions, such as believing that a whale is a fish [McCoy 84]. By classifying the types of misconception and why they occur, and associating answer schemas with these misconception types, she can generate the following sorts of responses:

I thought a whale was a fish.

No, it's a mammal. Although it has fins and lives in the water, it's a mammal since it is warm blooded and feeds its young with milk.

McCoy's schemas differ from McKeown's in several ways. Most importantly they encode *two* kinds of information — the content of the proposition and the *communicative role* of that proposition (such as 'concede', 'deny' or 'state'). So, a schema might include items like '(deny (classification OBJECT POSITED))'

[McCoy 86]. In this way her schemas are similar to the EDGE discourse plans, where communicative (discourse) and content level information may be similarly combined. However, as will be shown in chapters 4 and 5, the EDGE model allows the communicative role to influence whole sections of discourse and not simply how a single proposition is presented.

Recent work by McKeown has considered how the user's underlying goals should affect what information is included in answers to questions about course scheduling [McKeown *et al* 85, McKeown 88]. The user's goals may be inferred from the preceding dialogue, and may include things like 'wanting to major in CS', with possible subgoals of 'wanting to take AI' and 'wanting to take Databases'.

The user's knowledge, misconceptions and goals are obviously important in influencing the text. However, a complete picture has to include the speaker/hearer roles and the attitudes of each participant. For example, in a situation where the user is to feel dominant and confident, little new information should be included, the topic should be led by the user, and simple sentences should probably be used. In a teaching situation the way information is presented may be very different. Hovy uses rules such as these to generate narratives which depend on a number of aspects of the discourse situation [Hovy 85, Hovy 88b]. Such pragmatic factors influence both what to say and (especially) how to say it. Jameson, on the other hand, concentrates on the effect of speaker bias, generating biased descriptions of rooms depending on the expectations (and anticipated response) of the hearer, and the goals of the speaker [Jameson 83].

In summary, systems have been developed which generate text depending on the knowledge, goals, misconceptions, roles and attitudes of the hearer. All these may be important in a flexible model of explanatory discourse, but the most critical is arguably the effect of the user's domain knowledge.

2.1.1.4 Allowing Interaction

The majority of work on text/explanation generation ignores the interactive nature of communication and concentrates on generating single utterances. The exceptions view an ‘explanatory dialogue’ as involving textual explanations and follow-up questions. The Eurohelp project, for example, provides menus of follow-up questions after explanations, allowing some continuity and opportunity for clarification [Hartley & Smith 88]. Moore goes further and allows the dialogue context (the structure and content of the previous planned explanation) to influence responses to follow-up questions [Moore & Swartout 88b, Moore & Swartout 88a]. This includes disambiguating ‘why’ questions based on the discourse focus and user model, selecting *object perspective* ([McCoy 85]) for comparing two objects based on the user’s goal, and answering inarticulate follow-up questions by using the previous explanation plan to guess at problems with the explanation.

Another system which uses the dialogue context to influence responses is McKeown’s system for tailoring explanations and recommendations depending on the user’s goals — these may be inferred from the discourse [McKeown 88]. However, none of these systems consider interaction within an explanation¹, perhaps because of the limited length of the explanations involved.

2.1.1.5 Adding Meta-Comments

Another way textual presentation may be improved is to add ‘meta-comments’ — that is, comments which refer to the discourse itself, rather than conveying new content. Zuckerman and Pearl discuss how meta-comments may be used to improve mathematical explanations [Zuckerman & Pearl 86]. They classify meta-comments into three groups: *knowledge-organisational* meta-comments provide

¹Moore does in fact discuss this issue, assigning it to further work.

links between propositions depending on the conceptual relations between them; *knowledge-acquisition* meta-comments aid acquisition of knowledge by preparing the hearer for what follows; *effect-maintenance* meta-comments relate to the emotional effect of the text (e.g., discouraging). Meta-comments are used to improve a mathematical explanation by first planning the proof, then adding meta-comments of different types depending on the structure of the proof and the difficulties of the steps.

Knowledge organisational meta-comments appear to be similar to the discourse markers which may be used in RST-based generation systems, depending on the rhetorical relations between discourse segments. However, few of the other systems described use the other types of meta-comment.

2.1.1.6 Conclusion

In order to generate coherent text it is necessary to consider the overall structure of the text (such as the relations which may occur between sections of the text), how the text content is influenced by the user's knowledge, goals and attitudes, how interactions are managed (and influence the text) and how the presentation may be improved with discourse markers and meta-comments. Though there has been a limited amount of work in each of these areas, the area is still a very active one with many questions unanswered.

2.1.2 Curriculum Planning

Curriculum planning is concerned with the organisation of information at a much higher level. In the last few years there have been several attempts to generate systems that will plan an individualised curriculum given knowledge of the user's understanding in the domain. Many of the ideas in these systems also apply to generating explanatory text and discourse. Wenger, for example suggests that:

“A curriculum is quite clearly a plan; however, the concept actually applies to most didactic activities. Even a local explanation, for instance, can be considered a plan. Since an explanation rarely consists of a single conceptual elements, the granularity of communicable units will require an incremental presentation. Viewed as schemes to cause a transition between two knowledge states using a succession of intermediate steps, a non-trivial explanation is a plan, a kind of mini-curriculum, for leading the student along a local learning path.”
[Wenger 87, pg 397]

This section will therefore review recent work on curriculum planning, and summarise how it may be applied to explanation generation.

2.1.2.1 Planning to Change Knowledge States

One of the first attempts at generating an individualised curriculum was the planner developed by Peachey and McCalla [Peachey & McCalla 86]. This system used a simple ‘Strips’-like planner [Fikes & Nilsson 71] extended to allow partially ordered plans. Teaching operators (in the domain of micro-economics) were defined with prerequisites and effects — these effects included add- and delete-lists so operators could be defined to ‘remove’ some buggy concept or add some new concept to the student’s knowledge. The curriculum was made up of units which included procedures for checking whether the user had grasped the relevant concept, and the planner could be reinvoked in case of failure.

Peachey and McCalla’s work introduced many of the problems of curriculum planning. The effects of operators are necessarily uncertain — students may learn spontaneously or fail to learn what is being taught. Important issues are therefore how to check on the user’s knowledge, and how to replan when inconsistencies develop between predicted and checked knowledge.

2.1.2.2 Simple Skeletal Planners

Another approach to curriculum planning is to use a simple skeletal planner. Skeletal planners use prebuilt plan fragments which can be either specialised or refined into more detailed sub-plans [Friedland & Iwasaki 85]. In curriculum planning this means having a set of teaching operators (skeletal plan definitions) for different tutorial strategies and for teaching different skills. Russel, for example, has two sets of skeletal plans [Russel 88]: pedagogical plans define ways of teaching different skills, while strategic plans define different teaching strategies (such as ‘Say why you want to teach it, test, teach subgoals, teach prerequisites, teach concept body and question student’). Murray has four levels of planning — planning lesson objectives using domain-dependent knowledge, planning the presentation, planning to deal with interruptions, and meta-planning based on defined preferences for the planning process [Murray 89]. Murray considers issues such as how to mix opportunistic and planned approaches, dealing with student initiative in a planned curriculum — this issue was discussed earlier in [Derry *et al* 88] though in the context of a much simpler planning framework.

MacMillan presents another approach which essentially uses prebuilt skeletal plans [Macmillan *et al* 88]. His system uses a basic framework for instructional planning where domain-specific skeletal plans and models may be added. The basic planning approach is to incrementally refine the skeletal plans as required (though other strategies are possible within his framework).

These approaches show how multiple levels of skeletal planning can be used to plan a simple individualised curriculum. General tutorial and discourse instructional plans can be used with more domain-dependent plans to provide a flexible framework for developing a curriculum.

2.1.2.3 Generalising the Architecture: Blackboard Planners

One approach to making the plans more flexible is to use a general architecture where multiple knowledge sources may contribute to the planning process, given a global representation of the evolving plan [Macmillan & Sleeman 87, Macmillan *et al* 88, Murray 89]. In practice, these currently produce simple, multi-level skeletal planning as described above. However, within the basic architecture there is potential for exploring more flexible styles of dynamic planning.

Whether or not a blackboard architecture is used, having an explicit representation of the past and future plan is vital in dealing with the problems of replanning (cf. [Moore & Swartout 88b]) and dealing with interruptions.

2.1.2.4 Other Approaches

Another approach to curriculum planning is to have an explicit representation of curriculum relations (e.g., prerequisite, part-of, misconception) and heuristics for selecting next topic. Brecht and Elsom-Cook essentially take this approach [Brecht *et al* 89, Elsom-Cook 85]. Lesgold, on the other hand, treats each item in the curriculum as an 'object' which may pass messages to other (e.g., prerequisite) objects [Lesgold 88]. However, his approach appears functionally similar to the multi-level skeletal planning approaches, in this case using three levels: the meta-issue level, the lesson object level and the domain level.

2.1.2.5 Planning an Explanation

The above systems largely plan the high level structure of a lesson or sequence of lessons². The lowest level of Russel's system, for example, is the 'instructional

²Elsom-Cook's system is the only exception.

unit' — a page or two of text and graphics³. Yet as Wenger suggests, the same ideas apply to the generation of explanations. Breuker et al. have shown that people appear to be using plan based approaches when generating quite short pieces of advice (a few utterances) and use a skeletal planning based approach to structure help information [Breuker *et al* 87]. General strategies (such as 'announce, give context, new info, consolidate and evaluation') are specialised given a particular problem, and terminate in 'tactics'. There are a range of possible tactics which take specific objects as arguments, and have associated text templates which are used to present the information to the user.

The sorts of explanations which the EDGE system attempts to model fall somewhere between the short pieces of help of Breuker et al. and the larger sections of curriculum of Russel, Macmillan and Murray. They are perhaps more like the information seeking dialogues in the 'GUS' system [Bobrow *et al* 77]. This system used a hierarchy of frames to structure the dialogue and to provide a context for interpreting answers, extra information and questions. However, where the GUS system aims to find out what the user wants, the EDGE model applies where the system wants to communicate some information.

A long explanation should be constructed co-operatively, so the planner must be able to deal with interruptions from the user. The basic unit for communicating knowledge will therefore be the exchange (rather than the 'page of text' [Russel 88] or the speech act [Breuker *et al* 87]). However, the architecture developed in the context of the EDGE system has many similarities with the above systems.

³One reason why planning approaches have started with the curriculum level is possibly the existence of explicit plan-based approaches in the educational psychology literature on which some of these systems are based (e.g., [Gagne & Briggs 78]). The planning of explanations is more subconscious and less explicit.

2.1.3 Conclusion: Organising Content

The two approaches described above were both concerned with the organisation of information to be conveyed. They are currently quite distinct disciplines. However, as text generation research becomes concerned with generating larger (multi-page) sections of text, and becomes increasingly interested in dialogue, and as curriculum planning ideas become applied more to shorter sections of presentation, the two may begin to converge. The sort of explanatory discourse which this thesis is concerned with must combine ideas from both fields.

2.2 Structuring Dialogue

From work on text generation and curriculum design we can see how content may be structured which depends on the knowledge and goals of the participants. However, both these areas fail to consider the interactions which may take place as a long and complicated piece of textual information is presented to a user. Work on text generation is based largely on written rather than verbal, interactive communication, and so fails to exploit the interactive capabilities of the computer, while work on curriculum planning generally terminates with large pre-specified interactional sequences.

This section will consider some approaches based on studies of human verbal interaction. These include studies of tutorial dialogues (and the resulting computational models based on the observed strategies) and linguistic studies of discourse structure. Both are important for generating explanatory discourse.

2.2.1 Tutorial Dialogues

Work on tutorial dialogues arose out of attempts to build 'intelligent tutoring systems' (ITSs) able to emulate some of the dialogue capabilities of human tutors.

Scholar: Approx what is the area of Brazil?
Student: 2,500,000 square miles.
Scholar: Wrong. Please indicate if the following statement is correct: the area of Paraguay is approx 47,432 square miles.
Student: What is the area of Brazil?
Scholar: The area of Brazil is approx 3,600,000 square miles. Now answer the question you didn't answer before.
Student: Incorrect.
Scholar: Very good. The area of Paraguay is approx 157,000 square miles. The language in Brazil is French. Right or wrong?
Student: Wrong.
.....
Scholar: What is Brazil?
Student: Tell me about Peru.
Scholar: Peru is a country. It is in South America. The population is approx 7,800,000 people. The capital is Lima. Now answer the question you didn't answer before.
....

Figure 2-1: Simplified Extract from Dialogue with 'SCHOLAR' System

Human tutorial strategies were analysed, and these strategies implemented as a computational model. The work has progressed from a simple mixed initiative question-answer system with no sophisticated strategies, to systems using multi-level control to select relatively complex tutorial strategies.

2.2.1.1 Early Systems: Scholar

The first computer system which aimed to emulate human dialogue capabilities was the 'SCHOLAR' system [Carbonell 70]. This system managed a question-answer teaching session on South American geography. A simple mixed initiative dialogue was possible, with students able to interrupt and ask questions of their own (illustrated in figure 2-1, taken from [Carbonell 70]).

In the first version of the system, topics were selected largely at random, using

just a simple notion of relevance to the current topic based on the underlying (semantic net) domain representation. The system marked which topics had been covered in the knowledge base, and used this to guide later topic selection. This represented an early and very simple use of an 'overlay' user model (discussed in section 6.1).

The structure of the dialogue was very simple, consisting of an unstructured sequence of question-answer-response exchanges. It was clear that this did not correspond to the structure of human tutorial dialogues, so later work was based on empirical studies of expert human teachers.

2.2.1.2 Rules for Modelling Human Dialogues

Later work on the SCHOLAR project involved analysing a number of tutorial dialogues to abstract the rules that appeared to be used by human teachers (discussed in [Wenger 87, pg 37]). Some of these were implemented in a revised version of the system. For example, topic selection was improved by using a set of heuristic rules for selecting the next topic, and an agenda of topics to be covered. However, there were a number of features of human dialogues which were considered important, but which could not be easily incorporated into the framework. These included starting a topic with a phase of pure presentation when it was unfamiliar to the student, and reviewing topics later in the dialogue.

Another important landmark in work on tutorial dialogues is the 'WHY' project [Collins & Stevens 82, Stevens & Collins 77]. Again, this was based on the analysis of numerous human dialogues, but this time with the emphasis on formalising the rules used by teachers who used a 'socratic' approach in their teaching — leading students to realise the consequences of their own beliefs.

The result of this analysis was a simple tutor in the domain of meteorology. Scripts representing causal sequences of events in the domain were used as the basic representation of knowledge, with sub-scripts representing finer details of the sequences. These event sequences provided the global organisation of topics

in the dialogue, while 'socratic' rules guided the system's local response to the user's errors. These rules were based on human socratic strategies and particular properties of the chosen domain and types of question asked. One rule used was the following:

IF the student responds with insufficient factors
THEN formulate a general rule asserting that these factors are sufficient, and ask the student if that rule is correct.

One of the deficiencies of the WHY system was the lack of any kind of student model. The structure of the dialogue is a result of local responses to specific situations. However, using a limited set of socratic teaching strategies, the system could produce coherent dialogues which (unlike the SCHOLAR dialogues) could plausibly have been produced by a human tutor.

At about the same time, Clancey was developing the 'GUIDON' system [Clancey 83a] — a tutoring system based on the 'MYCIN' expert system. In this system the student was presented with a medical case to diagnose. The global structure of the dialogue was therefore dictated by this consultation, while local interactions and topic selection were again controlled by a set of production rules based on observations of human tutors. The system was able to infer what domain rules the user knew based on the questions asked in the consultation. This was recorded as an overlay model on the expert systems rule base, which indicated whether there was evidence that the user knew the rule, could apply it, or had actually used it.

2.2.1.3 Strategies for Tutorial Discourse

The systems discussed above all fail to capture the high level strategies used by tutors, being based on production rules which fire in specific local circumstances. Woolf attempted to remedy this by using a network based on three

levels of discourse control [Woolf & McDonald 84]. The top level was the pedagogic level, which involved deciding whether to introduce, tutor or complete a topic. The next level was the strategic level, involving decisions such as how to repair a misconception, or how to explore the user's competency. The final level was the tactical level, which included methods for giving questions and acknowledgements, or for suggesting analogies or examples.

In the original system the dialogue was controlled by traversing this network in a fixed sequence until a 'meta-rule' fired, when control was passed to another state and a new sequence. The original intention of the meta-rules were to capture changes of strategy used by teachers when some approach begins to fail. However, in the system implemented they were used for more routine local decisions. The architecture was therefore later reworked in terms of discourse 'schemas' [Woolf & Murray 87]. These schemas, based on a modified 'augmented transition network' (ATN) formalism [McDonald 86], defined the range of options given a particular teaching goal. For example, given the goal to provide some remediation, the schema defines that there should be some 'grain of truth acknowledgement' followed by one of three types of remediation depending on the type of error.

Woolf's work is important in introducing the idea of multiple levels of control in guiding tutorial discourse, and more recently in suggesting the use of schemas in controlling tutorial discourse. However, the work does not consider the selection of topics and has only a very simple notion of user model.

2.2.1.4 Topic Selection

Most of the above systems have paid little attention to the planning of the dialogue content. Topics have been selected either by traversing the underlying knowledge representation, or by making decisions based on the immediate context. None of the systems have considered the overall organisation of topics

based on explicit curriculum information about which topics are prerequisites of others.

Planning of how to order topics has generally been the concern of work on curriculum planning, described in the previous section. These systems are driven by specific goals, rather than opportunistically teaching according to the immediate situation. A tutoring system (and also an explanation) will normally need both types of control. The teacher will have some goals concerning what must be got through in a session (or explanation) if it is to succeed, yet will take advantage of circumstances to correct misconceptions and introduce other material.

2.2.2 Linguistic Theories of Discourse Structure

Work on tutorial dialogues (discussed above and in [Wenger 87]) has a lot to say about local responses to particular situations, but less to say about the global organisation of the content or dialogue structure. This section will briefly review three approaches to the linguistic (or sociolinguistic) analysis of discourse structure. The first (discourse analysis) emphasises the common structures of particular discourse types, and includes detailed work on the structure of classroom discourse. Rules are derived (based on the analysis of numerous example dialogues) which capture possible sequences of utterance types almost as a sentence grammar captures possible sequences of syntactic categories. The second (conversation analysis) is concerned more with the local organisation of casual conversations, deriving from detailed analysis rules which apply to all human verbal interaction. The final approach (used in most AI work on discourse) emphasises the fact that discourse results from plans and goals, and looks at the structure that results from this.

These reviews aim to be illustrative rather than comprehensive, discussing those aspects which are of particular relevance to explanatory dialogues and the EDGE system. Each field covers a wide area with much ongoing research.

2.2.2.1 Discourse Analysis

Work on discourse analysis essentially applies ideas derived from work on the structure of sentences to multi-sentence discourse. Utterances are generally classified according to their function in the interaction and constraints hypothesised about possible (and impossible) sequences of utterances. Some kind of 'grammar' of possible utterance sequences is constructed, and further examples checked to see if they conform to this grammar.

The Structure of Classroom Discourse

Sinclair and Coulthard (who pioneered this approach) chose formal classroom discourse as a suitable domain which might be amenable to this sort of analysis [Sinclair & Coulthard 75]. The theory developed is sufficiently well formalised to be adapted for computer control and interpretation of discourse, and a version ([Burton 81]) has been used in several dialogue systems [Ferrari & Reilly 86, Wachtel 86]. Wachtel, for example shows how such a model influences pragmatic constraints on the discourse.

Sinclair and Coulthard suggest that classroom discourse may be analysed at five levels. The first of these is the *lesson*. A lesson consists of a number of *transactions* (such as informing or directing on some topic). Transactions in turn consist of a number of *exchanges* of different types. In particular, they begin and end with *boundary* exchanges. Each different type of exchange has a structure based on the types of moves made by the participants. For example, a *teacher-elicited* exchange consists of an *initiating* move on the part of the teacher, a *response* move from the student, and a *feedback* move from the teacher. These moves may again be made up of a number of linguistic *acts*. For example, a feedback move may consist of an *assessment*, an *evaluation* and a *comment*.

Classroom discourse was analysed by classifying utterances and sequences of utterances according to their function in the tutorial dialogue. This was

somewhat subjective, but it was claimed that a number of researchers came up with the same classification. The structures were particularly clear at the levels of exchanges, moves and acts, while higher levels (e.g., the ordering of transactions) could not be easily analysed in these terms, depending more on the pedagogical goals of the teacher in a particular situation. Also, sequences of informing or commenting statements were not analysed, being treated as a single act. The analysis therefore dealt with very different issues from those considered in work on tutorial dialogues in the field of 'Intelligent Tutoring Systems'.

Subsequent Work on Discourse Analysis

Subsequent work on discourse analysis has concentrated on generalising the theory and giving it a firmer methodological foundation. Intonation was found to be an important feature of discourse, providing an objective key into the structure and classification of moves and acts. Work on exchange structure was generalised to other types of discourse (e.g., [Burton 81]) and detailed accounts given of other discourse genre (e.g., [Harris 88]). The importance of non-linguistic acts became clear, in the growing field of kinesics. Several of these developments are described in [Coulthard & Montgomery 81].

Some of the recent work has rejected simple hierarchical models as being too restricted to cope with the complexities of exchanges in less formal discourse. For example, Fawcett, Mije and Wissen use a complex model based on a network of possible moves and choices [Fawcett *et al* 88]. This type of model seems largely consistent with the work on conversation analysis, discussed in the next section. Carlson goes further in abandoning simple hierarchical models, claiming instead that there are a set of rules which govern our use of language (*dialogue games*), rather than rigid structures [Carlson 83]. These rules may reference any items in the dialogue history, including those implicit in the dialogue rather than explicitly stated. Reichman, on the other hand, accepts that conversation is hierarchically organised, but develops a complex model based on networks of

choices of conversational moves, and relationships between the *context spaces* of the moves [Reichman 81].

Other work has looked at how discourse models relate to discourse genre and 'register'. Ventola presents an approach (systemiotics) based on three levels. Discourse genre uses register as its expression plane, which in turn uses language [Ventola 88]. It is therefore the discourse register (e.g., roles, attitudes, domain and mode of interaction) which have a direct influence on discourse structure.

A more concrete example of the influence of discourse genre on structure is given in Harris's work [Harris 88]. She showed that, while high level structure (sequences of transactions) was fixed for a general discourse type (judicial court)⁴ how these transactions were realised (in terms of allowed sequences of exchanges) depended on the more specific discourse sub-genre, and further how the linguistic acts were realised depended on the attitudes of the speaker.

There are clearly no simple rules for structuring language. However, for relatively formal types of discourse the simple hierarchical discourse models proposed by Sinclair and Coulthard and developed by Burton [Burton 81] provide a computationally tractable basis for dialogue control and interpretation.

2.2.2.2 Conversation Analysis

The other current approach to the analysis of human interaction is *conversation analysis*. While discourse analysis is based on linguistic models, and is concerned primarily with deriving formal representations of discourse structure in constrained domains, conversation analysis is concerned more with how ordinary casual conversation is managed. It views conversation as being locally managed, and would reject theories based on global plans. Conversation analysis

⁴The sequence of transactions which she claims is common to a wide range of courts is 'Summoning, Swearing in, Preliminary, Information Gathering, Ordering and Closing'.

provides many useful insights into the management of dialogue (summarised in [Levinson 85, ch. 6] and [Suchman 87]), and can accommodate a wider range of dialogue phenomena than the formal, normative rules used in work in discourse analysis. In this approach the structure of the discourse emerges out of local interactions in particular situations, and is not imposed by high level 'grammatical' rules. There are preferences or expectations, rather than rules, which guide the interpretation of utterances — for example a question provides an expectation that an answer will follow, but does not demand it.

This approach seems to provide a better basis for the analysis of human conversation, and perhaps for human-computer systems of the future. However, the very lack of formal rules and categories, and its dependence on face-to-face interaction (such as the importance of pauses and intonation) makes the approach hard to apply directly. On the other hand, the normative rules from discourse analysis may be acceptable where there is one party with a dominant role and where the discourse is highly structured.

The following sections however will consider some of the insights from this work, especially where they may be applied to the computer control of interactions.

Turn Taking

One of the early concerns of conversation analysts was how people manage turns at conversation. Turns are managed so that control passes between participants with virtually no detectable pause between utterances. Sacks, Schegloff and Jefferson (who dominated early work in the field) suggested that there were predictable 'Transition Relevance Points' (TRPs) in conversation, where control could switch to another party. These might be after a word, phrase or sentence, but could be predicted in advance by the other party [Sacks *et al* 78]. At such points, a simple rule system would determine who would talk next. If the speaker explicitly selected another person, that person would speak next. Otherwise,

anyone could come in. If no other person contributed, the first speaker would continue. These rules emphasise the interactive nature of even continuous talk — there may still be TRPs where another speaker could contribute, and even the absence of comment represents a contribution to the dialogue.

Adjacency Pairs

Another feature of local conversational organisation is how utterances provide expectations for subsequent utterances. So, a question will provide an expectation that an answer will follow, and the following utterance (whatever the surface form) will tend to be interpreted as an answer to the question. Such pairs (e.g., question-answer, greeting-greeting) are referred to as adjacency pairs. It is useful to think of the structures derived in discourse analysis in these terms. Elements in an exchange provide strong expectations for subsequent elements, but do not require them absolutely. Utterances will be interpreted in terms of these discourse expectations where possible.

Insertion Sequences and Side Sequences

Utterances in an 'adjacency pair' need not necessarily be contiguous. For example, a question may be followed by a clarification question and answer pair before being replied to. These are referred to as insertion sequences. More general types of interruptions are side-sequences, where a topic is interrupted to be later resumed. Jefferson considers how topics are resumed in conversation [Jefferson 72].

Opening and Closing Sequences

Another feature of conversations is how they are opened and closed. Common sequences of utterances are used, effectively to negotiate the beginning and end

of a conversation. For example, telephone conversations frequently end with a four part sequence such as:

S1: OK
S2: OK
S1: Bye
S2: Bye

The 'pre-sequence' of 'OKs' is used to negotiate the end, while the 'Byes' are used in leave taking, giving a final opportunity for either party to continue the conversation.

Conversation Analysis and Discourse Analysis

The types of structures discussed in discourse and conversation analysis are very similar. Both are concerned with exchanges and consider opening and closing sequences. However, while discourse analysts give descriptive accounts of formal structures observed, based on some type of classification, conversation analysts are concerned with how people locally respond to the conversational situation.

The main conclusion which can be taken from this work is that discourse has a hierarchical organisation. Even casual conversations involve opening and closing sequences, and have some structure within exchanges. However, it is unclear how far this organisation results from conscious planning, or as the result of reacting locally to particular goals and situations. Hierarchical models of discourse appear to work reasonably for formal types of discourse such as classroom or courtroom. Here there are strong constraints on what may follow what, dictated by the conventions of the situation and the social roles of the participants. However, in casual conversations far fewer of these constraints apply, and the organisation of the interaction is influenced more by the local situation.

2.2.2.3 Plan-based Models of Discourse

AI (computational linguistic) models of discourse have generally been based on the plans or intentions of the speakers. This section will review one established model of discourse structure ([Grosz & Sidner 85]) based on this approach, and show how a plan-based approach has been used in both understanding and generating dialogues.

Intentional, Attentional and Linguistic Structure

Grosz and Sidner consider three types of structure in discourse. *Intentional structure* is based on relationships between the intentions of the speaker realised in the text. For example, in task descriptions an intention to make the hearer know how to do some task may *dominate* the intention to teach a subtask, while a precondition of the task would result in a *satisfaction precedence* relation in the intentional structure. Intentional structure is linked to the *linguistic structure* of the discourse (its segmentation) and the *attentional structure* of the discourse. The attentional structure or focus space stack includes a representation of all the items in the discourse which are in *focus* at a particular point. This includes the objects and events being talked about and the current *discourse segment*. The focus space stack will include the focus space of the current discourse segment plus all the focus spaces of dominating discourse segments.

The following example text may be used to illustrate various features of the model. This example is taken from ~~one~~ transcripts of human explanatory dialogues. As in examples in later chapters, 'E' will refer to the expert, and 'N' will refer to the novice.

- (1) E: In this circuit we have an output whose voltage depends on the amount of light falling on the LDR.
- (2) And the more light falls on here, the more it will conduct so the higher the voltage will be here.
- (3) N: What does LDR stand for?

- (4) E: Light dependent resistor.
- (5) N: What's this bit?
- (6) E: That's a variable resistor, so you can change how the output level varies with the resistance of the LDR.

In this example, utterances 1-6 form part of a discourse segment whose principle purpose is to explain how a simple circuit works. Utterances 3-4 form a subsidiary discourse segment dominated by the larger one, as do utterances 5-6 and utterance 2. In utterances 5-6 the main item in focus is the variable resistor. However, in Grosz and Sidner's model the items in focus in dominating discourse segments are also considered — there is an implicit focus space stack based on the focus spaces of all the dominating discourse segments. So in this example 'this circuit' (a light-detector circuit) will also be in focus, so may be implicitly referred to in utterance 6 — 'the output level' refers to the output level of the light-detector circuit.

Grosz and Sidner discuss how these types of structure affect surface linguistic phenomena such as the use of anaphora, clue words and meta-comments. For example, clue words (or discourse markers) such as 'now', 'first' or 'anyway' mark the beginnings of different sorts of discourse segments. Pronoun use depends on the attentional state, and in particular how the *centre* of an utterance — the single most salient object — changes. The theory has since been used as the basis for generation systems which deal with such phenomena. Dale, for example, develops a theory of anaphora in simple task (recipe) descriptions [Dale 89] using a modified version of the theory.

Inferring Plans in Discourse

Several discourse understanding systems have attempted to derive some sort of *intentional structure* or plan from the user's utterances. Allen and Perrault, for example, consider how the speaker's goals may be inferred from simple train station dialogues, given knowledge of plans in this domain [Allen & Perrault 83].

Litman extends this to include plans which deal with domain independent discourse phenomena such as interruptions and corrections, with interrupting sub-dialogues put on a stack of discourse models or plans [Litman 85, Litman & Allen 84]. This latter approach (using both domain and discourse plans) will prove important in generating explanatory discourse.

The plan-based approach to discourse interpretation has also used the idea of a *dialogue game*. Wittgenstein first suggested that a dialogue could be treated as a kind of game, with moves, rules and objectives [Wittgenstein 68]. This idea was formalised in the early 70s by Levin and Moore [Levin & Moore 77] and by Power [Power 74] (discussed below). The basic idea in both of these was to formalise exchanges (= dialogue games) in terms of the effects they had on the speaker and hearer, and build this into a plan-based framework. However, where Levin and Moore were concerned with interpreting dialogues, Power was concerned with generating dialogues.

Generating Dialogues from Plans

Power showed how simple dialogues could be generated, based on the goals of the speakers. His system was later reworked by Houghton and Isard who extended the generation to consider the surface structure of sentences, providing a nearly complete, albeit simple model of language generation [Houghton & Isard 85].

Power invented a world inhabited by two robots. The world consisted of a door and a bolt, with the robots initially at either side of the door. Each robot had various capabilities, such as being able to see or to push the bolt, and various beliefs, such as believing that pushing the door opens it. A robot might begin with a goal, such as to open the door, and have to cooperate with the other robot in order to do this. The dialogue would result directly from this goal-driven behaviour.

There were two main components to the system. A simple planner could be used by each robot to reason how to achieve its goal. When the plan involved

something which the robot did not know or could not do, but which the other robot might know or do, then an exchange would be initiated. These exchanges might involve getting attention, making something known, finding something out or getting something done. Each exchange type (or *dialogue game*) was defined in terms of its effects, constraints on applicability and the type of reply expected (only two part exchanges were considered). So, a robot would plan to achieve its goal until it met an obstacle which the other could help with, then a dialogue game would be chosen and initiated. The other robot would recognise the game, so be constrained to reply appropriately (though could interrupt to check on the state of the world or to comment on the others plan). Following an exchange, the world view of each robot would be updated, and then the planning would continue where it left off. This would result in dialogue such as the one below, taken from Houghton and Isard (who extended Power's world to include multiple doors):

Fred: Doris
Doris: Yes
Fred: I want to be in
Doris: I see
Fred: Could you push the yellow door.
Doris: No, because the bolt isn't up.
Fred: Is there a bolt that is up?
Doris: No
Fred: How do you get a bolt to move?
Doris: You get to be in, then you slide it.
Fred: Could you slide the green bolt.
Doris: OK
Fred: Push the Door
Doris: OK

Of course, Doris isn't being very cooperative here. She could have recognised Fred's goal and volunteered to open the door for him⁵. However, the dialogue

⁵Power later reworks the approach to allow such cooperative behaviour, using the idea of mutually agreed plans being pursued by both participants [Power 87].

illustrates how different types of exchange (and a resulting simple discourse structure) may result from a single goal.

Another system where the dialogue is based on a complex planning process is Sacerdoti's hierarchical planning system [Sacerdoti 77]. The system creates a plan of how to do some complex task and presents a step to the user. The user can then ask a limited range of questions (such as asking how to do some step, saying that it can't be done or asking for the next step). The system can then use the plan network created to respond appropriately.

In less toy domains, structuring a discourse entirely from the underlying goals and plans becomes very difficult, requiring a good understanding of human memory and reasoning. It is not even clear that it is possible, as to some extent discourse structure becomes institutionalised (e.g., courtroom or classroom) and no longer driven solely by the goals of the participants. Yet the basic idea of dialogue games may be usefully carried across as a way of structuring individual interactions. Elsom Cook, for example, uses dialogue games to control interactions, but uses heuristics to order topics and topic specific routines to select appropriate dialogue games [Elsom-Cook 85].

2.3 Text and Discourse Structure in Explanatory Dialogues

A computer-based 'explanatory dialogue' falls half way between uninterrupted written text and face-to-face interaction. So far, most models have been based on one or the other (such as those described in the discussions above). This section will attempt to demonstrate that none of these existing paradigms provide an adequate model by themselves, and will point towards a framework which integrates a number of these ideas. The section will be organised around var-

ious issues considered important for explanatory discourse, and which will be developed further in the following chapters.

2.3.1 Structuring the Explanation Content

An explanation must be structured so that no concepts are used which the user doesn't understand (though new concepts may be introduced if based on more familiar ones), and so that the user can see the underlying structure and purpose of the discourse. Recent work on text generation (e.g., [Moore & Swartout 88b, Hovy 88a]) has considered how descriptions may be planned given knowledge of the affect of various 'rhetorical relations' and knowledge of the user. However, explanation generation should also make use of relations between concepts in the domain, as in the work on curriculum planning (e.g., [Peachey & McCalla 86]).

So, in expository discourse, what is needed is a planner which makes use of prerequisite and sub-skill relations in the domain, but which is still consistent with an established discourse theory (e.g., [Grosz & Sidner 85]). This allows an explanation to be planned based on some underlying goal and on the assumed knowledge of the user, and to be structured using clue words, meta comments and anaphora so that the user may see the intention and structure of the discourse.

2.3.2 Structuring Interactions

An explanatory dialogue is generally relatively well structured, both in content and interactions, with the explainer taking the dominant role. However, it must be possible to deal with problems in the dialogue as they occur. Users may ask questions, or the system may ask a question to check on understanding and reveal a misconception on the part of the user. A theory of explanatory discourse must therefore be able to plan the overall organisation of the dialogue (such as when to open a topic with a boundary exchange), as well as deal with interruptions.

Existing (ITS) systems generally deal with interruptions either by simply putting the interrupted topic on a stack, or by biasing the topic choices so that previously mentioned topics will tend to be returned to. Yet a full treatment of interruptions should consider whether to accept the interruption (was the system about to deal with the problem anyway?), and how to resume the interrupted discourse. Maintaining a hierarchical model of past discourse and any future discourse planned makes such reasoning possible.

The theory must be able to accommodate opportunistic responses such as those analysed in [Collins & Stevens 82] and [Woolf & McDonald 84]. However, the formalism used for selecting such responses should be consistent with the underlying discourse model chosen.

The theory should also consider the hierarchical structure of interactions, discussed in [Sinclair & Coulthard 75] but not currently used in controlling any dialogue system ⁶. The structures observed were independent of the domain being taught, so similarly it should be possible to separate this discourse planning level from the planning of the specific content. However, the particular model developed by Sinclair and Coulthard was based on the analysis of formal classroom talk with the teacher in control, interleaving questions with the presentation of information. It must therefore be adapted significantly if it is to apply to one-to-one explanatory dialogues.

In order to capture these different aspects it is necessary to combine a model based on the hierarchical organisation of discourse [Sinclair & Coulthard 75] with strategies for dealing with the user's errors and misconceptions

⁶It could be argued that such rigid formalisms are avoided with good reason, being based on formal classroom discourse where the student has little control. However, in explanatory discourse it may be acceptable for the explainer to retain control until the explanation is 'complete'. The model will at least be extended to allow interruptions and opportunistic responses.

[Woolf & Murray 87], and strategies for using the underlying discourse model being constructed to reason about interruptions. These should all be based on a consistent model of discourse structure. Grosz and Sidner's model is sufficiently general to accommodate all these aspects.

2.3.3 Inferring and Using a Model of the User's Knowledge

In an explanatory dialogue it may be important to tailor both the content and the way it is conveyed to the particular user. Ideally, the goals, knowledge and attitudes of the user should all affect the explanation. All of these have been considered in the work on text generation described above. However, the most important thing to consider is the knowledge of the user. If the explanation is at too high a level it will be not understood, while if it is over-simple it will be verbose and uninteresting.

Two approaches have been considered in work in text generation. Several systems filter the output to not tell the user things that they think they know. This is a useful basic principle, but over-simple — if you don't tell users anything they know then arguments will be incomplete and the cognitive load on them will be too great. Another approach is to use different explanation strategies depending on the knowledge of the user [Paris 87]. This is an important idea, but needs to be generalised beyond the simple choice of two description strategies used by Paris.

Work on tutorial dialogues has generally concentrated on local responses to errors and misconceptions, rather than the use of a general user model. However, the user's assumed knowledge may affect the choice of next topic, and their general level of understanding or confusion may affect the dialogue or remediation strategies chosen. Work on curriculum planning on the other hand gives a principled way of selecting topics depending on prior knowledge. If prerequisites of a topic are unknown, then they will be taught. This seems a better approach than to simply 'miss out propositions that the user already knows'.

So, the user's knowledge should affect the planning of the explanation, including what strategies are selected and whether prerequisite information is included. However, it should also affect the presentation. In general this might include the use of referring expressions, and decisions about whether to include a lot of directive meta-comments. However, in this thesis the user's knowledge will only influence decisions about whether to question a user on a topic, or just to inform the user. Even with this simple decision it is possible to create more flexible dialogues with phases of pure presentation as suggested in the SCHOLAR project.

Of course, principles for tailoring discourse to a user's knowledge are useless if that knowledge about the user cannot be inferred and updated⁷. Very few existing systems both infer the user's knowledge from the dialogue, and use it to influence it ([Chin 88] and [Clancey 83a] are two exceptions). No system influences dialogue choices, explanation strategies and prerequisite information as well as being able to infer and update all the necessary knowledge about the user.

2.4 Conclusion

This section has briefly reviewed some of the work on text generation, curriculum planning, tutorial dialogues and two 'flavours' of discourse analysis. It showed that none of these approaches is sufficient in itself to generate explanatory discourse of the type considered here. A combined approach is necessary, which considers formal linguistic theories and plan-based generation as well as the more flexible opportunistic approaches in work tutorial dialogues. The approach must consider the planning and organisation of the content of the explanation

⁷A review of work on inferring a model of the user's knowledge is given in chapter 6.

(reviseable based on new information about the user), the organisation of the dialogue, and how each interacts with a model of the user's knowledge. Chapters 4, 5 and 6 consider each of these aspects, and develop many of the points in this chapter in more detail. However, before moving on to such detailed description of the EDGE system, the next chapter will discuss some of the work that motivated the approach and provides the basis for the specific implementation — analyses of human explanations and dialogues.

Chapter 3

Analysing Human Explanations

In this chapter an analysis of human verbal explanations will be described. There are many possible ways to analyse text and dialogue, from word use in different text types to plan-based analyses of conversation. This thesis is concerned primarily with the structure and content of explanatory discourse, so it is analyses of text and dialogue structure which are most relevant. The approaches used are based on the work on text generation, discourse analysis and tutorial dialogues discussed in the last chapter. Where possible, my results are compared with and augmented by results from these other sources.

Of course, there are many other ways of examining explanations. For example, one can look at the type of questions asked, or consider what the underlying domain model might be. Some of these are considered in [Cawsey 88a].

A corpus of verbal explanations of how simple electronic circuits worked is used for these analyses, with occasional illustrations taken from the task dialogues used by Grosz and others and given in [Deutsch 74]. It is important to consider how far such analyses will generalise to other types of explanation. The chapter will therefore begin with a discussion of different types of explanation which might be analysed. The following two sections will concentrate on describing different types of analysis and applying these to one particular class of explanations. The final section will summarise the results.

3.1 Types of Explanation

There is a wide variety of possible sources of explanations, and many ways an explanation can vary. Not only can the domain be different, but the roles and attitudes of the participants will influence what is said. Equally important for human-computer interaction is the *mode* of interaction — results valid for verbal or written communication may not apply to interacting with a computer. These factors are discussed further below.

Subject Matter

There are several different types of explanation, as well as a wide range of possible domains. For example, explanations of how to do complex tasks, explanations of how something works, and justifications all have different structures which have been examined by different researchers (e.g., [Grosz 77] (tasks) [Paris 87] (descriptions/how things work) [Weiner 80] (justifications)). This thesis will concentrate on explanations of how physical systems work, but explore how some of the same principles may be applied to task descriptions.

Roles and Attitudes

The roles and attitudes of the participants also effect the structure of the explanation. For example, explanations between friends and colleagues will be different from explanations between teachers and students, or explanations in court. This thesis assumes a teacher-student role — however, this role may be temporarily assumed in a particular situation (such as when friends provide tuition in some topic). It also assumes friendly attitudes between participants, and no deliberate ‘putting in place’ by asking difficult questions. While this is the

role assumed in the analysis, the framework developed is generalisable to other roles and attitudes.

Mode of Interaction

Explanations may be analysed based on a range of modes of interaction. At one extreme, natural verbal explanations may be recorded without the participants knowing. Radio phone-in programs provide a rich source of this type of explanation, though in a somewhat artificial situation. Pollack, Kidd and Gilbert all examine this sort of explanation [Pollack *et al* 82, Kidd 85, Gilbert 87a]. Telephone conversations also have the possible advantage that there is no visual contact, therefore corresponding more closely to human-computer interaction. However, in human-computer interaction the human may observe the system's graphical actions, for example. Phone conversations therefore provide an inadequate model of explanations where such graphical actions (sketching and pointing to diagrams) are important.

Telephone conversations allow the speaker to use intonation to indicate discourse structure and focus. One way of eliminating this is to analyse explanations carried out across computer terminals. Visual contact may be eliminated, or at least controlled. Grosz, for example, uses a fairly complex arrangement whereby the novice communicates verbally to the experimenter, who communicates to the expert via a computer terminal [Deutsch 74]. In this experiment the expert can see what the novice does, but the novice cannot see the expert. Schuster, on the other hand (analysing cooking explanations) uses a simple terminal link, where each speaker is asked to signal the end of a turn by saying 'OK' [Schuster 83b]. In the Eurohelp project (e.g., [Winkels *et al* 88]) there is a simple terminal link, but (like Grosz) the expert is allowed to see what the novice is doing. All of these, then, attempt in some way to emulate the mode of human-computer interaction. However, none allow the novice to see what the expert is doing. Help

and explanation systems of the future will involve interactive text and graphics, so it is important not to eliminate this when exploring human explanations.

The most restrictive mode of discourse is written text. Here there are no opportunities for interruption, and no use of intonation or pauses to indicate discourse structure. There may be some reference to diagrams, but this is a fixed diagram rather than a dynamic model. This is the mode examined by most researchers in text generation (e.g., [McKeown 85, Paris 87]), yet is far more restrictive than human-machine communication allows.

The examples in this thesis are based on face-to-face verbal interaction in an experimental setting. This has the disadvantage that intonation and gestures are allowed, yet the advantage that the speaker may use pointing (to a diagram) and sketching to aid the explanation. In retrospect the 'ideal' mode of interaction to use would be where the communication was via computer terminals, but where the expert was allowed to sketch and point, and even run simulations and get graphs drawn from them.

Immediate Discourse Context

As well as the general properties of the discourse discussed above, the immediate discourse context will influence the explanation. This will include the discourse goals motivating the explanation. The EDGE explanation corpus was obtained in an experimental situation with the experimenter requesting the explainer to give a particular sort of explanation. It is therefore largely decontextualised. Although this makes it easier to analyse, it is important to be aware of this limitation. Encyclopaedia and manual descriptions also often have little immediate discourse context. Results abstracted from analyses of this sort may not be appropriate when generating explanations in a particular context, such as within a consulting dialogue or advice session.



Range of Examples

In order to make general statements about explanations, and in particular about the effect of mode, domain or roles on explanation structures, a large corpus of explanations is needed. In practice however, different researchers have examined different types of explanation, generalising from different experts and novices and from different examples. The result is a characterisation of particular types of explanation. However, it is hard to generalise from this to other types.

If a comparison is to be made between different modes of interaction or different levels of novice understanding, it is important to keep other factors fixed. For example, the same expert should ideally be used, or alternatively, a sufficiently large sample of experts so that the results may be generalised.

The analysis below is based on explanations of seven expert-novice pairs, and four experts giving uninterrupted explanations. Each explained from four to eight circuits and the explanations averaged one (typed) page long when transcribed. From this one can make some generalisations about this type of explanation (from a corpus of 32 monologues and 52 expert-novice explanatory dialogues). However, many features which we would like to analyse only occur in a small number of these explanations. General statements cannot be made about these — they can only be used as examples of phenomena which exist, and which an explanation system should be able to deal with. The corpus is also not large enough to make general statements comparing different experts, novices or circuits, though guesses can be made concerning how these factors affect the explanation.

Summary — Type of Explanation Used

In conclusion then, there is a whole range of types of explanation which could be analysed. A comprehensive theory would have to consider how each aspect (domain, mode, roles and attitudes) affects the resulting discourse, requiring a

wide range of analyses. In order to generalise, a large number of explanations of each type should be examined.

The analysis in this thesis is based on one type of explanation, which may be summarised as follows:

Domain: How a range of simple circuits works.

Mode: Face to face, verbal, recorded but not videoed. Access to circuit diagrams (but not real circuits) and pencil and paper.

Roles: Expert-novice roles assuming teaching objectives. Roles set by experimenter by asking experts to explain to novice how the circuits worked.

Attitudes and goals: Friendly, positive attitudes with the goal of teaching or learning how the circuits worked.

Immediate discourse context: Experimental setting with no other motivating goals.

Range: Seven different expert-novice pairs and four experts explaining up to eight circuits each.

3.2 Text Structure: Schemas, Grammar Rules and Plans

This section will consider a number of approaches which have been used for analysing uninterrupted text, and show how these have been applied to explanations in the corpus described above. Most of these approaches are based on classifying the clauses in a text in some way, and (given this classification) describing common structures of a range of texts. Structures may be described using *schemas*, *grammar rules* or *plans*. A single example explanation will be

-
- (1) Well, this here looks to be a potential divider circuit
 - (2) and it contains two components
 - (3) one of which is a light dependent resistor
 - (4) and the other is a variable resistor.
 - (5) The purpose of this circuit is to provide a varying output voltage on this line here.
 - (6) The idea is that when light falls on the light dependent resistor its resistance changes.
 - (7) So therefore because the resistance of this series circuit has changed, the voltage at this point here can change.
 - (8) So when light falls on the LDR the resistance of this part of the circuit is very low.
 - (9) And so this voltage here is a high voltage.
 - (10) In darkness the resistance of the LDR is large
 - (11) and so the voltage at this point here drops to a low value.
 - (12) So we can get a high or low output depending on how much light is falling on the light dependent resistor.
-

Figure 3-1: Example Uninterrupted Explanation

used in the discussion and is given in figure 3-1. This example was a verbal explanation given by a physics teacher to the experimenter.

3.2.1 Classifying Utterances

The first step in the analysis is to classify the different utterances in the explanation. The classification must be general enough so that a range of explanations may be classified using the same categories, yet constrained enough so that it may be used for selecting specific utterances in generation. For example, using categories such as 'introduce', 'summarise' or 'elaborate' appears to be insufficiently constraining, while a category such as 'identify-LDR' would only apply to circuits with an 'LDR'. More appropriate types of category may be taken from McKeown's work.

McKeown classifies utterances from a set of 20 *rhetorical predicates* [McKeown 85, pg 23-25]. However, it turns out to be quite hard to give any useful analysis based on this set. For example, how should line 5 be classified? There is no rhetorical predicate for an object's *function* so it would have to be classified as an *amplification*.

The analysis below therefore uses McKeown's predicates where possible, extending the set where necessary to include similar predicates. Such classification is necessarily subjective, especially where new categories are 'invented'. However, it provides a useful starting point for the analysis of text organisation.

- Identification. *This is a potential divider circuit.*
- Constituency. *And it contains variable resistor.*
- Function. *The purpose....*
- Process. Lines 6-11 (based on cause-effect clauses such as line 8).
- Conclusion/Cause-Effect. *So we can get*

Even this simple ordered classification captures a range of other explanations. Consider for example the following less coherent explanation, also taken from the corpus:

- (1) Right, potential divider,
- (2) thermistor temperature dependent, resistor.
- (3) So, the output is going to vary between what, I can't think, the..., so the numbers are difficult, but what you're going to have, of the order of, you're going to have between.. Basically the output is going to vary with temperature..
- (4) So if the resistance on that one drops, that means we're going to get a higher output.

This too consists of an identification, constituency, function (or cause-effect?) and a conclusion.

There is some ambiguity and overlap in these classifications, especially where different 'cause-effect' clauses serve different functions in the explanation. However, it serves to indicate typical sequencing of explanations, and the sorts of things included — it is clearly much more than a simple trace of device behaviour. One important thing to note is that the ordering of the explanation corresponds to the stages of scientific explanation discussed by Cox [Cox *et al* 88]. The first three of these stages are the stage of figurative knowledge (what it is and its properties), the functional stage (what it does/what it's for) and the cause-effect (process) stage. Understanding each of these is considered a prerequisite for the next.

3.2.2 Describing Structure: Schemas

McKeown classified a wide range of descriptions and produced a set of four schemas which captured all the different structures observed in these textual descriptions. There is considerable flexibility within the schemas, and ambiguity in the classifications, so it is unclear how strong a result this is. An example schema is given in figure 3-2.¹

None of the schemas given quite captures the sorts of structures observed in circuit explanations. However, they were designed for descriptions rather than explanations of how something worked, so perhaps this is not surprising. Paris [Paris 87, Paris 88] uses a combination of a *constituent* schema and a process trace to generate explanations/descriptions of physical systems. She also extends the schema to include 'identification' as the first item. The resulting schema (including decision points) is given in figure 3-3.

¹Note that "/" indicates optional alternatives, "+" indicates that the item may appear 1 to n times, "*" indicates 0 to n times and "{}" optionality.

Constituency
 Cause-effect* / Attributive* /
 { Depth-identification/ Depth-attributive
 { Particular-illustration/ Evidence}
 { Comparison/ Analogy} }+
 { Amplification/ Explanation/ Attributive/ Analogy}

Figure 3–2: McKeown’s Constituency Schema

Identification (intro to superordinate)
*If there is no local expertise for superordinate, do **Process Trace** of
 superordinate*
Constituency (description of subparts)
For each subpart, do:
*If there is local expertise do **Depth Identification** else do **Process**
Trace
Attributive*

Figure 3–3: Paris’s Revised Constituency Schema

Explanation Schema

{ { Identification } { Constituency } { Function } } /
{ **Analogy** }
{ **Process** }
{ Cause-effect* } (or summarise)
{ Identification } (speaker recognises device class at end)

Constituency Schema

{ Constituency }
{ Component-Identification/ Component-Attributive }*
(optional attributes of components — behaviour/function)

Process Schema

For one or more significant input values:

For each causal link:

Optional identification (e.g., this is an amplifier so)

Either cause-effect or **process**

Analogy Schema

Comparison

Cause-effect (of different component)

Figure 3-4: Explanation Schemas

Paris also defines a *process-trace* schema which is used in conjunction with the constituency schema. The process trace allows recursive calls to explain subpart processes and optional descriptions of parts involved. The combination of these two schemas gets close to capturing the structure of the circuit explanations, yet still does not include a causal trace of the behaviour of the whole device. It is therefore necessary to define a schema based on analysis of the circuit explanations. A schema which captures our 32 uninterrupted verbal explanations is given in figure 3-4 (using schema recursion on the items in bold).

This schema characterises all of these explanations, with one exception where

constituency is given before *identification*. Every stage in the explanation is optional, yet the schema imposes strong ordering constraints on the explanation.

It should be clear how the example explanation in figure 3-1 fits this schema. The 'identification, constituency, function, process, cause-effect' sequence identified before fits the top level explanation schema. Lines 2-4 in the example follow the **constituency** sub-schema and consist of a 'constituency' description ("it contains two components") followed by the identification of two components ("one of which is a light dependent resistor and the other is a variable resistor."). Finally, lines 6-11 follow the process schema by giving two cause-effect links for each of three input values (if you allow 'varying' to be classified an input value).

3.2.3 Describing Structure: Grammar Rules

Another way to describe the possible explanation structures is to use a set of 'grammar' rules². Rumelhart uses a context free grammar to describe the structure of simple stories [Rumelhart 75]. In a similar manner, grammar rules may be used to describe explanations, as shown in figure 3-5.

In these rules, each component is optional. The rules only impose constraints on ordering. In McKeown's schema's too there was generally only one compulsory element for each schema. One might expect that a process trace would be compulsory in explanations of how things work, but in practice some explanations gave the background information (function, constituency etc.) and summarised the behaviour, assuming that the causal trace could be inferred from this.

This representation (grammar rules) has the advantage that it begins to capture the hierarchical organisation of the discourse, and allows the structure

²A set of grammar rules may have the same representational power as a single schema, but tend to capture the hierarchical structure of the discourse. When schema recursion is used the two representations become very similar.

How-it-works \Rightarrow Description, Process, Summary.
Description \Rightarrow Identification, Constituency, Function.
Description \Rightarrow Analogy.
Constituency \Rightarrow Structure, Component*
Component \Rightarrow Identification, Attribute*.
Analogy \Rightarrow Comparison, Component.
Process \Rightarrow I/O-process+
I/O-process \Rightarrow Causal-step*
Causal-step \Rightarrow Identification, Causal-link.
Causal-step \Rightarrow I/O-process.
Summary \Rightarrow Causal-step*.

Figure 3-5: Grammar of Explanations

to be described at different levels. However, it still doesn't explain why such structures exist or provide a clear link with the overall goal of the speaker.

3.2.4 Explaining Structure: Plans and Discourse Models

This section will consider two approaches where the discourse structure may be linked with the overall goal of the speaker. The first is the approach used in this thesis — using sub-skill and prerequisite skill relations in planning operators. The second is a currently popular approach to text generation — using rhetorical relations between text sections as planning operators.

3.2.4.1 Sub-skills and Prerequisite Skills

As was shown earlier, the ordering of items in the explanation is not coincidental. It is consistent with the stages of understanding used by Cox [Cox *et al* 88] (figurative, functional and cause-effect). Other constraints include needing to

be familiar with a device's structure and components before being able to follow a causal explanation. There are therefore prerequisite relations between the skills in the domain. In a similar manner the hierarchical discourse segmentation, captured in the grammar above, may correspond to sub-skill relations between concepts being taught. For example, knowing what a light dependent resistor does might have the sub-skills of knowing what it does when it's light and knowing what it does when it's dark.

If each rule in the discourse grammar is rewritten as a planning operator, making the prerequisite and sub-skill relations explicit, then this has several advantages:

- It is consistent with Grosz and Sidner's theory of discourse structure, which explains a number of linguistic phenomena. The plan structure provides a discourse model which explains discourse segmentation and focus.
- By explaining the schemas, it provides a better basis for generating explanations in other domains without such exhaustive analysis.
- It provides a better basis for tailoring discourse given knowledge of the student's understanding — if a prerequisite skill is already understood, it need not be included in the explanation.

Of course, the fact that explanation generation will be modelled as a planning process is not to claim that human explainers necessarily have any knowledge of planning or of the relationships between concepts. The human explainer may have abstracted ways of explaining or describing from many examples and use those abstracted schemata when explaining — yet it may still be seen as based originally on some planning process, and be usefully modelled as such.

The planning operators used are discussed in chapters 4 and 7 and given in appendix B. The range of possible plans which can be generated should in

```
(defplan how-it-works (device)
  :preconditions ((structure (device)))
  :subgoals ((teach causal-behaviour (device))
             (teach what-it-does (device))))
```

Figure 3-6: Top Level Plan for Explaining Circuits

principle correspond to the schema in figure 3-4. Figure 3-6 illustrates a simplified version of the top level plan used. The plan corresponds to the first line in the grammar in figure 3-5. The precondition *structure* corresponds to the 'description' but is only given when not already understood. *causal-behaviour* and *what-it-does* correspond to the process and summary parts of the grammar rule.

3.2.4.2 Rhetorical relations

If we can identify 'rhetorical relations' between sections of the explanation (like 'compare-contrast' or 'motivation-enablement') then it may be possible to use these relations as planning operators, as in [Hovy 88a, Moore & Swartout 88b]. However, as Mann and Thomson observe [Mann & Thompson 87] the sorts of descriptions and explanations considered by McKeown [McKeown 85] (similar to those considered here) make relatively little use of such rhetorical relations — most of the relations would have to be encoded as versions of the 'elaboration' relation, which includes describing attributes or identification. Ordering conventions are partially built into the planning operators used by Hovy by allowing 'growth points' in the relations where extra relations may be initiated. However, while the relations ensure some overall coherence, domain dependent constraints are not included. They are therefore very useful for organising unstructured information, but not where other constraints (related to the hearer's understanding and ability to assimilate complex new information) dominate.

3.2.5 Conclusion

This section has shown how the organisation of the content of simple explanations of circuit behaviour may be analysed and a schema developed which captures the observed structure. This was rewritten as grammar rules, capturing the hierarchical structure and segmentation of the discourse, then as plan operators, capturing the relations between skills being taught. These will provide the basis for the planning of text content described in the next chapter.

The actual schema was derived from a relatively small sample in a limited domain. However, the structure was clear enough to show that it did not fit any of McKeown's schemas (though it was similar in many respects).

This thesis does not aim to present another detailed analysis of a simple domain, but rather to present an approach to generating expository discourse which will generalise to a wide variety of domains and types of discourse. Hierarchical structure may be observed in all expository text, and it can generally be considered in terms of relations between the ideas or skills being conveyed. This should be clear in the structure of a thesis. In principle, early sections should provide the introductions necessary for the reader to understand later sections. One chapter may convey a single main idea, while being composed of a number of sections discussing different stages or aspects of that idea. Text is frequently planned by deciding first on the overall structure (the approximate content of chapters or sections), then recursively filling in sections. It is this approach to text planning and structuring which is advocated for the computer generation of expository discourse.

3.3 Dialogue Structure

The above discussion has concentrated on analyses of the structure of monologue and written text. However, this thesis is concerned with explanations which involve interactions with a novice. This section will therefore be concerned with analysing features of the 'explanatory dialogues' obtained involving an expert explaining to a novice. Most of the results appear independent of the particular domain, but may depend on the roles of the participants in the particular discourse genre analysed.

The analyses consisted of initially studying the dialogues to detect common patterns and features, and examining related work. From this it was possible to abstract hypotheses about dialogue structure which could be tested. The dialogues were then re-examined to confirm or refute particular hypotheses. Much of the analysis is somewhat subjective. The section will therefore include numerous examples so that they may be judged independently.

3.3.1 Dialogue as Interrupted Text?

If we are to carry across results from the last section, it is necessary to analyse a number of dialogues to see if the underlying explanation has the same structure. The explanations analysed varied greatly in style, including:

- Explanations dominated by the expert, with few interruptions.
- Experts 'leading' the novice through an explanation, but expecting most of the steps to be supplied by the novice.
- Short explanations from the expert, followed by numerous follow-up questions from the novice.

- Expert and novice cooperatively constructing an explanation, each contributing steps and querying the other. In this case the roles of ‘expert’ and ‘novice’ become somewhat confused. This type of dialogue may arise when the two participants have comparable levels of expertise.

However, despite these radically different styles of dialogue, all 52 dialogues (by seven expert-novice pairs) could be viewed in the following terms. The dialogue involved traversing a schema or plan (such as described in section 3.2.2) apart from interruptions, and follow-up questions. Steps in the schema could be contributed by either novice, expert or via an exchange involving both.

This is not a very strong claim, due to the flexibility of the schemas, and subjectivity in interpretation. Both the classification of utterances and decisions about boundaries of interruptions are perhaps disputable and in some cases interruptions and follow-ups dominated the dialogues. Also, to capture all these further examples the schema would have to be extended to allow more complex constituency and process descriptions, such as allowing additional circuit or component attributes in descriptions³. However, if we accept that the schemas in section 3.2.2 represent normative patterns of explanation rather than all possible explanations, then we should not expect every explanation to fit perfectly.

If we accept that the dialogues can be viewed as following the same sorts of schemas as uninterrupted (verbal and written) explanations, then this suggests that Grosz’s claim that dialogue structure mirrors the structure of the underlying task [Grosz 77] may be successfully generalised to other domains, where the underlying domain knowledge is less structured, but where the explanation itself is highly organised. This is a controversial claim — many would suggest that the structure of the dialogue is an emergent property, and not based on any preconceived plans. Gilbert, for example shows that advisory dialogues (on

³These extensions would appear to be consistent with McKeown’s and Paris’s models.

claiming benefits) do not fit any fixed schema [Gilbert 87a]. In these dialogues the control is equally balanced between advisor and advisee, and there are fewer constraints on the order of the advice. Gilbert suggests that in these cases it is local coherence which dominates the dialogue structure, and suggests that after each utterance by the advisee, the advisor seeks to find a partial plan which is coherent with the last utterance, yet contributes to the overall goal of the conversation. This seems to be an appropriate model for many types of conversation. However, in the explanations analysed the ordering constraints and dominance of the explainer allow the simpler model based on schema or plan traversal with interruptions.

The following examples (figures 3-7 to 3-11) illustrate a range of dialogue types, each of which can be seen as following part of the explanation schema⁴. Each of the five dialogues was generated by a different expert-novice pair.

In figure 3-7 we have an example where a minimal explanation is given by the expert (function-process), followed by follow-up questions from the novice (about 15 question-answer pairs in the full example). The initial explanation follows the schema, yet misjudges the user's knowledge, so fails to give the necessary prerequisite information (such as a constituency description). Most of the follow-up questions are concerned with eliciting this prerequisite information. This perhaps emphasises the importance of dialogue in the absence of a detailed user model (as suggested by [Draper 87] and others).

The example in figure 3-8 is similar in many respects, but involves interrupting questions from the novice. Again, the questions (this time interruptions rather than follow-up) seem to be concerned with obtaining missing prerequisite information (component description and device function). The overall schema

⁴In all these examples '....' indicates sections of the dialogue which have been omitted, 'E' and 'N' represent the expert and the novice. The complete explanations from which these extracts were taken are given in appendix A.

E: In this circuit we have an output whose voltage depends on the amount of light falling on the LDR. And the more light falls on here, the more it will conduct so the higher the voltage will be here.

N: What does LDR stand for?

E: Light dependent resistor.

N: What's this bit?

E: That's a variable resistor, so you can change how the output level varies with the resistance of the LDR.

.....

Figure 3-7: Dialogue 1: Follow-Up Questions in Explanation (Light Detector)

which appears to be being traversed is 'constituency-process', though the process explanation is in terms of circuit equations (applied backwards from the output value sought) rather than input-output behaviours, and the constituency description is more complex involving additional circuit/component attributes. Note that the expert's third utterance starts with a clue word ('well') and is essentially a repeat of part of the first utterance. This is a standard technique for resuming an explanation, so confirms the hypothesis that the intervening dialogue be regarded as an interruption, though the continued plan may be revised based on knowledge acquired about the user, in this case going into more detail.

In these first two example dialogues, it is the expert explaining, and the novice asking questions. Figure 3-9 illustrates another type of dialogue — here the expert is leading the novice through the steps of the explanation. Again the 'schema' which is being traversed in 'constituency-process' where the constituency description concentrates on describing features of the transistor.

In this example (3-9) the expert is still controlling the progress of the explanations, asking very directed questions. However, the last two examples illustrate

E: These components here, you might consider them as being both resistors. Two variable resistors. I can write down a relation for resistance..

N: You'll have to tell me what a resistance is.

E: A resistor is just

N: I see, or at least, I think I see.

E: Well, in this circuit here there are just two resistors, switched in serial mode. You can think of it as one resistor supplied with a voltage 9V, and there will be a current flowing through the circuit. Well, the output can be calculated by, when you know the current flowing here then you can assume that the output doesn't take any current, so the current here is nil. The output voltage is determined by this current multiplied by this resistance. In order to calculate the current you have to include this resistor in your network as well,

N: Why are we bothered about what the current through there is. What's going to happen, I have to know the purpose of it.

E: In the end you want to know this voltage

N: OK

E: I can just write down a relationship, this is I_1

Figure 3–8: Dialogue 2: Interrupted Explanation (Light Detector)

E: OK, do you remember anything about transistors? Basically, the important thing about transistors is so the current going through the collector and emitter

N: Is amplified.

E: is 100 or 200 times bigger than the current going through here. The arrow here just tells you which way you should connect it.

Right, so what's going to happen here, say if you consider this as an input and this as an output, what happens when you make this voltage high, what happens to this?

N: When you make it high you probably get all the current from this end amplified.

E: Well, when you make the current high

N: It opens the gate.

E: Yeah, what's going to happen to the resistance between here and here?

N: It's going to be low.

E: So what's going to happen to the voltage here, if you have a very low resistance?

N: It's going to tend to a high limit.

E: Careful. If you let the current through these two then it's just as if you wired these two up, a little bit. So it's just as if the transistor weren't there and these two were connected. So if these two are connected there's nothing here in fact, so what's the voltage here going to be?

N: Well, the same voltage as the other end.

E: Yes, so it's going to be low.

.....

Figure 3–9: Dialogue 3: Guided Explanation (Inverter)

cases where the control is more balanced. In 3-10 the expert's role is to comment on the novice's explanation steps, while in 3-11 the explanation progresses cooperatively⁵. These types of dialogue are beyond the scope of the EDGE system, requiring unconstrained natural language input. However, they illustrate important types of discourse which should be captured by a general theory. This style of tutorial explanation is perhaps the most pedagogically sound, and best able to demonstrate the novice's understanding.

3.3.2 Exchanges and Interaction

The examples given above have illustrated the overall structure of the explanatory dialogues, showing how they could be seen as planned explanations with interruptions. Many of the interruptions and follow-up questions seemed to be concerned with accessing missing prerequisite information resulting from the expert wrongly assessing the novice's knowledge⁶. Later explanations in the same session might include such information, suggesting that the expert reassesses the novice's level of understanding as the session progresses.

In this section a more detailed style of analysis will be considered. Within sociolinguistics there has been a considerable amount of work on the structure of exchanges in conversation. In the last chapter one particular model was considered

⁵In this example the dialogues started with normal expert-novice roles with the expert contributing most steps in the explanation, but metamorphosed into more cooperative discussions requiring more from the novice. This same phenomenon has been discussed by [Leudar & Antaki 87], and could perhaps be seen as an instance of the 'cognitive apprenticeship' approach to learning, where the expert gradually supplies less and less support in some task (such as explaining) [Collins *et al* 86].

⁶This was also the case with Grosz's task dialogues, for example, in the fragment 'E: Install the pump. N: What is the pump?'.

E: Right, the next thing is exactly analogous really

N: OK, what's this?

E: That means it's a variable resistor

Right, the heat detector unit is exactly analogous in fact. How do you think it works?

N: Right, presumably depending on the amount of heat here, if it has, the more heat it has the more resistance it has?

E: Probably the less.

N: The less heat it has the more resistance it has?

E: Sorry, no, the hotter it is the less resistance.

N: The hotter it is the less resistance. So, if its not hot at all, if it's cold, if it's very high resistance therefore this middle one is closer to zero, and if its very very hot it has low resistance so its closer to 9V here.

But I don't see what difference this 5K or 10K makes?

.....

Figure 3–10: Dialogue 4: Novice Explanation with Feedback (Heat Detector)

.....

N: It would be an inverter if there was a wire coming out here.

E: Oh, right, but we'll have, that would be raised to zero volts.

N: Right, if there is light shining on this, the resistance goes down, which means that this will get to 9, and that's already 9, so this will be zero, that will be zero and because we have a zero here this is..

E: Up there, that thing up there.

N: We have a zero on the first one.

E: Well, the first one will be going from zero to one all the time.

N: So, looking at the table, that point there, for this whole inverter will be going 1-1, so it will always be on..

(Experimenter clarification)

So we've got 1, so this is always going to be at 9, so there's always a voltage across there,

E: So it's not going to produce a sound, so this is one where if you break the light then it will..

N: So that's what happens in this case so this will be varying between 9 and 0 so will be going nee naw nee naw.

Figure 3-11: Dialogue 5: Joint Explanation (Burglar Alarm)

Boundary exchange: (frame move) (focussing move)
Frame move: marker
Focussing move: (marker) (starter) meta comment/conclusion
Marker: discourse marker such as 'OK', 'Well'.
Meta comment: statement referring to future discourse.
Conclusion: statement referring to past discourse.
Starter: statement directing attention to a topic.

Figure 3-12: *Boundary Exchange Structure*

— that of Sinclair and Coulthard (at Birmingham) [Sinclair & Coulthard 75]. It might be expected that much of that model would carry across into the explanatory dialogues analysed — they too are 'tutorial' explanations with the explainer having a dominant role. However, the actual social roles (outside the explanation context) are balanced, and the dialogues are one-to-one. This section will therefore examine a number of example exchanges within the dialogues to see how they fit the Birmingham model. It will start with a discussion of *boundary* exchanges which are used to open and close topics, then discuss the various types of exchanges which go on within a topic.

3.3.2.1 Boundary Exchanges

Sinclair and Coulthard suggest that topics are opened and closed with 'boundary exchanges'. The structure of these is given in figure 3-12 (where brackets indicate optional items). This definition is sufficiently flexible to allow a wide range of initial and final utterances (or none at all). All the initial exchanges in the circuit explanations fit this model⁷, but not the final boundary exchanges.

⁷Though there were frequently no verbal opening statements. This may have been influenced by the experimental situation where a third party set the topics to be discussed.

-
- (1) What I'm going to do is to go through a set of circuits that I've got in front of me and I'm going to explain what the circuit does in a very general way and see if you can explain what happens when I change certain parameters. That way you'll be able to show your understanding of some simple circuits and some rather complex circuits.
 - (2) This is going to be a bit tricky and I'm going to get you to help me explain this one.
 - (3) What I'm going to do is to get you to explain this last circuit to me. Before I do that I better say briefly what a comparator is.
-

Figure 3-13: Opening Meta-Comments

Some opening statements are illustrated in figure 3-13 and figure 3-14. The first example (3-13) illustrates the use of meta-comments to open a topic⁸, while the second illustrates the use of discourse markers. All the speakers sometimes used these to signal a new topic, and some used them consistently.

The examples in figures 3-13 and 3-14 only consider opening exchanges. In general there were less explicit indications of topic endings. However, figure 3-15 illustrates some examples of closing exchanges. The important thing about these is that (contrary to Sinclair and Coulthard's model) the final initiative is always left with the novice. In many cases this was done using pauses in the conversation, yet in all the examples illustrated there were explicit checks or indications that the novice was happy and ready to continue to the next topic.

Within the main topics (explaining how circuits worked) there were sub-topics, such as describing a component, or starting the process explanation. These too were frequently indicated with discourse markers, confirming hypotheses about the hierarchical structure of the discourse. It seems likely that such

⁸All the 'meta-comment' examples came from the same speaker.

Speaker A:

- (1) OK, Right, this is a circuit where the output is from a voltage divider.

Speaker B:

- (1) Well, it's a similar story here
- (2) OK, do you remember anything about transistors?
- (3) OK, another one. So this is basically
- (4) OK, do you remember what this was.
- (5) Now, there's various bits here.
- (6) OK, this basically tends to be
- (7) OK, low heat detector.
- (8) OK, so what's this going to do?

Speaker C:

- (1) OK, the lines are wires
 - (2) Right, the next thing is analogous really.
 - (3) Right, now we get to the real fun, an inverter.
 - (4) Right, this thing is complicated.
 - (5) OK, well a comparator is fairly obvious.
 - (6) Right, low heat detector.
-

Figure 3–14: Discourse Markers on Topic Openings

N: OK
E: Is that sufficient?
N: I think I know what's going on with it.

N: OK, that's just what I wanted to know to start with.
E: OK?
N: Yep.

N: Yes, OK, I think I've got it.

N: OK

N: Alright.
E: Enough of that one.

N: That's enough of that one.

E: That do?

Figure 3–15: Final Boundary Exchanges

markers are especially important where the mode of communication is restricted, and intonation and pauses unavailable, however, there is no evidence here that this is the case.

In Grosz's dialogues, opening and closing exchanges have a different structure. At the beginning of a dialogue there may be sequence which serves to negotiate roles and procedures and to check on communication channels⁹, while the final exchange functions to check that the task is completed. Figure 3–16 illustrates some of these.

In conclusion then, Sinclair and Coulthard's notion of boundary exchanges

⁹This latter function does not often arise in face-to-face dialogues. Levinson discusses how opening exchanges in phone conversations serve to check on the communication channel and on the identity of the participants [Levinson 85, ch6]. Again we see how the mode of communication affects the dialogue structure.

Opening Exchanges

E: Hello, are you there?

N: Yes, great.

E: Good.

N: Ready here.

E: Good, Install the pump.

E: Hello?

N: OK, Go ahead.

E: Good morning, I would like for you to reassemble the compressor.

N: Alright, I'm supposed to just start doing that and ask any questions?

E: Yes, I may ask you what you're doing as we go along.

Closing Exchanges

E: OK, that's all. We're done folks. Everyone OK?

N: I am.

N: Looks to me as if it's working.

E: Good, it's working. We're done kids.

N: Thanks, we're done.

E: That completes the task. Thankyou.

N: Thankyou.

Figure 3–16: Opening and Closing Exchanges in Task Oriented Dialogues

at the beginning and end of topics seems to hold, though the detailed structure of these exchanges will depend on the type of discourse. At the beginning of an interaction (rather than a topic) there may be further opening comments to check on the communication and negotiate the dialogue procedure. In the circuit explanations, discourse markers and meta-comments are often used to open topics, and the ends of topics are negotiated. The closing boundary exchange does not fit in with the Birmingham model and corresponds more to the negotiations of ends of conversations discussed in [Levinson 85, ch6] for example. This may be due to the different roles of the participants, or due to the fact that the discourse is one-to-one.

3.3.2.2 Expert Questions

Having considered how topics are opened and closed, the next thing to consider is the structure of the exchanges within the topic. These include exchanges where the expert (teacher) asks the novice a question (*teacher-elicit* exchanges in Sinclair and Coulthard's notation), exchanges where the novice asks the expert a question (*pupil-elicit* exchanges) and *informing* exchanges. In this type of discourse it is reasonable to make an explicit distinction between the first two, as the roles of the participants are very different.

First, expert questions will be considered. Sinclair and Coulthard suggest that in classroom discourse they have the structure given in figure 3-17¹⁰. Again, as it stands this is a sufficiently flexible definition to allow a wide range of exchanges. The only compulsory acts are the *elicitation* and the *reply*. However some exchanges clearly don't fit this pattern, so the framework is extended to allow *bound initiations* which serve both a feedback and initiation function. For example, the phrasing of a question may imply that the previous answer was

¹⁰See [Sinclair & Coulthard 75] for detailed definitions of the different acts.

Teacher-elicitation exchange: Initiation (eliciting) move, Response move,
Feedback move.
Initiation move (eliciting): (marker) (starter) elicitation (prompt).
Response move: (acknowledge) reply (comment).
Feedback move: (accept) (evaluate) (comment).

Figure 3-17: Teacher-Elicit Moves

correct or incorrect. This allows structures such as 'initiation response bound-initiation response feedback'.

Figure 3-18 shows an exchange sequence with its associated (possible) structure. Even in this short example it is clear that there is some ambiguity in the analysis. For example, what exactly counts as a comment or a bound initiation? So, though it is possible to make this sequence fit Sinclair and Coulthard's model that may not be saying very much. Overall, while the model is useful in suggesting an overall hierarchical structure, and suggesting some of the moves and acts which might be included in exchanges, some of the details appear somewhat arbitrary and ill-defined.

To consider this type of exchange further we need to analyse what sort of responses (e.g., socratic teaching strategies) are used when novices get answers to questions wrong. These will be considered in section 3.3.4. However, the basic model that will be used is to have complex feedback strategies which may be used. So, an error might initiate a strategy involving another exchange. This might be viewed as a *bound initiation* as it serves as a follow-up to the previous exchange. If the answer is correct, or there is no suitable strategy, then a simple feedback structure will be defined, such as to assess, evaluate and rephrase the answer.

E: So what will be the voltage here?	Initiation (Elicitation)
N: 9V?	Response (Reply)
E: It will be closer to 9V, that's right.	Feedback (Comment?, Accept)
E: And if there isn't any light on it?	Initiation (Elicitation)
N: It will consume, I don't know how much.	Response (Reply)
E: OK, If no light falls on it then the resistance?	Bound Initiation (Marker, Elicitation)
N: Will be high, and will not let through current.	Response (Reply)
E: So the voltage here will be?	Bound Initiation (Elicitation)
N: Tending to zero.	Response (Reply)
E: Closer to nought.	Feedback (Accept)

Figure 3–18: Expert-Novice Questions

3.3.2.3 Novice Questions

The second type of teaching exchange considered involves the novice questioning the expert. According to Sinclair and Coulthard, this has similar structure to the expert questions, but without the feedback move. However, this type of analysis fails to capture the fact that questions may initiate long and complex explanations involving more exchanges. It seems vital to be able to treat these complex answers as responses, while allowing for exchanges within the response¹¹. Consider the example given in figure 3–19. In this case the expert's response to the initial question goes on for several pages or minutes, involving many further exchanges. In figure 3–20 (this time from Grosz's dialogues) again the novice's question dominates several pages of subsequent dialogue.¹²

¹¹This is equally true for expert-novice questions. Both the novice's response and the expert's feedback may involve a complex dialogue.

¹²The examples are indented according to my intuitions about the dialogue structure.

N: What are the components in that circuit?

E: Right, that one is a variable resistor, Its resistance can be set by that knob.

N: What's the difference between this variable resistor and the one in circuit B, because they have different diagrams.

E: There is a resistor between here, and this point is in contact with

N: Right.

E: This is an OPAMP, a feature of which is, well it has two significant features.

E: One is that the inputs make very little

N: I don't understand.

E: This circuit, this wire here (1 page)

....

So if the resistor here is set with the two halves equal then the voltage will be exactly half way between the two, so it doesn't make any difference, and similarly this one here has no effect, very small amounts come in.

E: The other thing that it does is amplify

Figure 3–19: Complex Replies and Resumptions

N: How do you get the flywheel off the pump?

E: Show me the belt and flywheel.

N: This is the belt. This is the flywheel and this flywheel seems difficult to remove from an axle that turns it. Have you got that?

E: Yes. You have to remove the belt before you remove the flywheel.

N: Do I have to remove the wheel from the motor and the belt in order to remove the belt?

E: No, first loosen the motor bolts

Figure 3–20: Novice Questions in Task Dialogues

The example in figure 3–19 also illustrates clearly how interruptions are frequently put on a stack, and the previous discourse resumed once the query has been dealt with (as discussed in section 3.3.1)¹³. According to Grosz and Sidner, such resumptions are frequently marked with discourse markers such as ‘anyway’. This would be particularly likely if the interruption was on a different topic. In the example explanations, such discourse markers were occasionally used, but not consistently. More often, pauses are used to mark the discourse structure.

3.3.2.4 Informing Exchanges

In Sinclair and Coulthard’s model, the other type of teaching exchange is the informing exchange. This involves an informing statement by the ‘teacher’, and

¹³Of course, it is not always appropriate to resume an interrupted topic. This is one limitation of the EDGE model which assumes that the main explanation should be resumed, although the detailed content of the resumed topic may be influenced by the change in the user model during the interruption.

optional acknowledgement from the novice. The informing statement could involve an arbitrary number of utterances.

In the explanatory dialogues analysed here, informing statements came from both expert and novice, and they frequently were followed by corrections or follow-up comments. Within an informing section, pauses and gestures could be used to provide 'acknowledgements'. A more useful model of informing exchanges can therefore perhaps be taken from theories of turn taking ([Sacks *et al* 78]). Here, there are various points in a person's turn where control may pass to another conversational participant. Failing to take up that opportunity may be treated as an implicit acknowledgement. So, in this approach, an utterance is generated ending with a pause. If the other participant does not come in here, then this is treated as an acknowledgement and the current speaker continues. Otherwise the turn ends and a new turn (and hence exchange) begins.

There is some question about how long the 'utterances' should be before a pause (or 'Transition Relevance Point' — TRP¹⁴). In face-to-face conversation they frequently occur at sentence boundaries, yet in HCI larger units may be appropriate. For example, current systems frequently present a page of information and allow some limited interaction after each page break. One advantage of using sentence boundaries is that the propositional content of expert-novice questioning exchanges, and informing exchanges may be the same, so a schema/plan may be traversed by expressing each proposition as either a questioning exchange or an informing exchange.

¹⁴Note that in face-to-face conversation there need not actually be pauses at TRP's. Intonation and eye movements may be used to signal these points.

3.3.2.5 Other Exchange Types?

There are two other types of exchange which could be considered. First, there are cases where the expert or the novice gives a direction or request. These included requests from the novice like:

- (1) Why don't you write down formulas so we can see what's going on.
- (2) I need to know what's the point of this particular bit, rather than an explanation at a lower level.

The other type of exchange consists of questions from the expert about what the novice knows and some signal from the novice whether they understand or not. For example:

- (1) First of all, do you remember from physics what these things do?
- (2) Have you got the hang of these potential dividers yet?

3.3.2.6 Conclusion

So, how useful are sociolinguistic models of discourse structure for describing explanatory discourse? It appears that while the basic approach is useful (such as considering how topics are opened and closed, and the characteristic structures of the different exchanges and moves) it is necessary to extend the model if it is to usefully capture the structure of the discourse. In particular, the following extensions are required:

- If discourse 'moves' are defined functionally (such as 'response' or 'feedback') then they must be allowed to span a whole section of discourse, perhaps involving further exchanges. Even when they are defined interactionally (such as 'predicted' or 'predicting' further moves [Stubbs 83]) then it would seem that a whole complex response sequence can be seen as 'predicted' by the previous utterance.

- A wider range of exchange types were used in this slightly less formal type of tutorial discourse. Each of these must be defined.
- The model should be extended to deal with interruptions. The structure of an interruption can be characteristically defined in terms an opening (push) marker, interrupting text and a closing (pop) marker and repetition or meta-comment to resume.
- In Sinclair and Coulthard's model, a monologue by the teacher is viewed as a single informing exchange. Considering work on turn taking, I believe that a more useful model is to consider each proposition conveyed as an informing exchange, and failure to interrupt or query as an implicit acknowledgement.

It is perhaps not surprising that the model needs considerable modification in order to capture useful features of the discourse. It was designed to capture the characteristic structure of interaction in a particular discourse genre, and not for computational modelling. The next section will consider how another theory of discourse structure introduced in the last chapter — Grosz and Sidner's model of intentional, attentional and linguistic structure — applies to the example dialogues.

3.3.3 Topic Structure and Focus

Grosz and Sidner suggest that discourse segmentation and attentional state (focus) may be explained by considering the intentional structure of the discourse [Grosz & Sidner 85]. In turn, discourse markers and the use of referring expressions relate to the discourse segmentation and attentional state. Analysing a discourse therefore involves using such clues to infer the discourse structure, while using the hypothesised discourse structure to resolve ambiguities in referring expressions. Litman showed further that the discourse could be analysed

in terms of discourse plans and domain plans, where discourse plans manage interruptions, for example [Litman 85].

In section 3.2.4. it was suggested that the content of the explanations could be explained in terms of plans, using prerequisite and sub-skill relations, while in section 3.3.2. the dialogue structure was shown to be hierarchical (figure 3-19). This suggests a plan-based hierarchical model of discourse structure involving discourse and content plans. If this is correct, one would expect some corroborating evidence in the data in terms of discourse markers and the choice of referring expressions.

Consider for example the use of pronouns in figure 3-19. In the last line 'it' refers to the operational amplifier and not to the resistor mentioned in the previous utterance. This is consistent with the discourse segmentation indicated (by indentation) and with Grosz and Sidner's theory of discourse structure.

The other main indication of discourse structure is the use of discourse markers. We have already noted how topics frequently begin with a discourse marker (such as 'OK'). In some cases, sub-topics are also signalled in this way. One expert for example, in a long explanation, signalled the start of the whole explanation with 'OK', and the start of the *process* section with 'Right'. Another speaker started the explanation with a meta-comment, and used markers to indicate the beginning of the *function* and *process* descriptions. There are fewer clear examples of discourse 'pop' markers — this is more often signalled using pauses or a repeat of some of the previous discourse segment.

An exhaustive analysis of the discourse in these terms is outside the scope of this research. However, the few examples noted seem consistent with the hypotheses that the discourse can be explained in terms of discourse and content plans and Grosz and Sidner's model of discourse structure. The plans will derive insights both from Sinclair and Coulthard's analysis and from Litman and Allen's model of discourse and domain plans [Litman 85, Litman & Allen 84].

3.3.4 Reacting to Misconceptions

The analysis so far has ignored pedagogical issues of discourse control considered in much of the work on tutorial dialogues. The WHY project, for example, analysed the strategies used by teachers who used the 'socratic' approach to teaching [Collins & Stevens 82]. When a student got an answer wrong, a limited number of strategies could be used to lead the student to realise her misconception and work out the correct answer. Woolf considered how different types of dialogue moves could be selected depending on strategic and pedagogical goals [Woolf & McDonald 84]. This included structuring questions and responses as well as responding to misconceptions.

The EDGE model does not attempt to capture the range of pedagogical strategies which may be used in tutorial dialogues. However, the framework should be able to support complex responses to misconceptions. In the example explanations, only one expert used interesting pedagogical strategies to lead the novice to the correct answer¹⁵. One of these dialogue sections was illustrated in figure 3-9 (with the complete explanation in appendix A) and another is given in figure 3-21 along with comments suggesting the basic strategies used.

In figure 3-21 two strategies are apparent. If a question is answered with a contributing step (rather than the final answer) then the expert acknowledges that step and prompts for further steps. If the answer to the original question is incorrect or confused, then the expert may use the strategy of prompting for sub-steps to lead the novice to understand the answer. Another strategy seen in figure 3-9 is to supply supplementary information (such as substeps or an analogy) and then ask the question again.

¹⁵Note that they had been given the goal of explaining, and not of teaching, so it is not surprising that most of the dialogues were 'interrupted explanations' rather than complex tutorial dialogues.

E: What's going to happen to the voltage here, depending on whether the light falls on it or not?	
N: When the light falls the resistance will be, uh..	<i>Contributes part of a step towards the answer.</i>
E: The resistance will be low.	<i>Provides answer to that step.</i>
N: So all the current will go through.	<i>Provides next step and pauses.</i>
E: OK, so there will be more current going through, so what will be the voltage here?	<i>Prompts for next step in argument and hence answer to original question.</i>
N: 9V?	<i>Provides answer (or very close).</i>
E: It will be closer to 9V, that's right. And if there isn't any light on it?	<i>Response and new question.</i>
N: It will consume, I don't know how much.	<i>Unhelpful answer/step.</i>
E: OK, if no light falls on it then the resistance?	<i>Simpler question leading to answer.</i>
N: Will be high, and will not let through current.	<i>Two correct steps.</i>
E: So the voltage here will be?	<i>So next question leads to answer to original question.</i>
N: Tending to zero.	<i>Correct answer (but perhaps badly phrased).</i>
E: Closer to nought.	<i>Rephrase answer.</i>

Figure 3-21: Tutorial Dialogue

The strategies chosen may depend on the type of question asked, on what the user knows, and on the particular pedagogical or dialogue strategy being pursued. This thesis does not attempt to enumerate the range of possible strategies and when they might apply. However, the discourse framework developed should be able to incorporate these kinds of strategies. Therefore two of the strategies mentioned (going through substeps and giving an analogy) will be formalised in the model.

3.4 Making Choices (and other issues)

It should be clear that both textual content and dialogue actions should depend on the expert's assessment of the user's knowledge. However, it is hard to deduce how it influences the explanation simply from the transcripts. It is impossible to be certain which choices depended on the user's knowledge, and which on the idiosyncrasies of the expert. To be able to make such deductions about how decisions were made, it would be necessary to get independent comments from the expert, either as the dialogue progressed or following the dialogue¹⁶. However, in this thesis decisions about how the user's knowledge will affect the discourse will be made on the basis of previous research and the author's intuition.

There are certain other problems with the analysis because of the mode of interaction (face-to-face, verbal) and the minimal transcription (pauses and intonation were not transcribed). This leaves a number of issues unresolved, such as how long an utterance should be before an acknowledgement is sought from the user. Also, physical actions such as pointing and sketching were not transcribed,

¹⁶Some approaches to deriving expert teaching knowledge have been discussed in [Goodyear 88] for example.

though it was generally clear when they had been used, both in asking questions about objects in the diagram, and in the main explanation. It is an open research question exactly how gestures such as these are coordinated with speech in human dialogue, but many researchers have emphasised their importance and suggested that they are linked to phrasal units in speech (e.g., [Kendon 83]). Others are working on how pointing gestures may be combined with natural language in human-computer interactions (e.g., [Neal *et al* 88, Schmauks & Reithinger 88]). In chapter 7 it will be shown how, to a limited degree, pointing and sketching is used in explanation output and pointing allowed in input in the EDGE system. However, there are many unresolved issues.

Another issue is the unnaturalness of the experiment, compared with a realistic teaching situation. The experts were given a definite goal (to explain how a set of eight circuits worked) and particular materials (a set of circuit diagrams and a pencil). It is not clear whether this is ever a pedagogically sound goal, or good materials to use when explaining circuits — why not use a real circuit? The experiments therefore say nothing about how to teach electronics — just about how this type of explanatory discourse is structured given a particular goal.

Despite these problems, the analysis revealed useful insights about expository dialogues, which could be developed computationally. These are summarised below.

3.5 Conclusion

This chapter has looked at a number of ways of analysing ‘explanatory dialogues’. First, the structure of the content was considered. Schemas were reviewed as a way of representing stereotypical ways of organising text content, but these had two problems. They typically don’t represent the hierarchical (intentional) structure of the discourse, and they fail to explicitly represent constraints on the

presentation of information based on sub-skill and prerequisite relations between concepts. Hierarchical plans were therefore advocated as a better representation.

Next, the structure of the dialogue was analysed. It was suggested that text schemas or plans also applied to dialogues, though there might be complex interruptions and follow-up questions which make that structure harder to see. If we look at text structure in terms of constraints on how people can assimilate information, then it is not surprising that this carries across to dialogues.

The structure of exchanges within the dialogue was analysed. Sociolinguistic theories of dialogue structure in classroom discourse were considered. While the concept of hierarchical dialogue structure (lesson, transaction, exchange, move and act) and different types of exchanges proved useful, the theory had to be significantly adapted to capture the structure of the dialogues. In particular, if moves were defined functionally as responses, for example, then they might span a whole dialogue section.

Grosz and Sidner's model of discourse structure appeared to apply well. Both the domain plans and the hierarchical structure of the dialogue could be viewed in these terms. This is similar to Litman and Allen's use of domain and discourse plans.

Finally, some simple tutorial strategies were analysed. It turns out that these too can be incorporated into the plan based framework, as a complex feedback sequence applicable to certain types of errors in certain situations.

The main contribution of the chapter is to show how multiple types of analysis referring to both content and interaction may be usefully applied to explanatory dialogues. However, the detailed results are somewhat tentative being based on subjective analysis of a relatively small sample of explanations. The next four chapters will show how this analysis has been realised as a complex computational model. The first three will consider the general problem of generating explanatory discourse, independent of the domain, while chapter 7 will look in detail at how dialogues similar to the ones analysed here have been generated.

Chapter 4

Structuring Communicative Text

One of the most important problems in the generation of explanatory discourse is how to structure the content of the explanation. The aim is to present the necessary information in such a way that the hearer may easily assimilate the new knowledge. In this chapter a model of text structure is presented which is applicable to the generation of explanatory text. The first part of the chapter will examine a number of principles for structuring text and present the criteria which a model should satisfy. This will be followed by a description of the text structuring aspects of the computational model developed.

4.1 Criteria for Generating Communicative Text

Before developing a model of text generation it is important to consider what principles govern the organisation and selection of text content. Many of the ideas may be taken from text linguistics, which is concerned with how texts function in human interaction [deBeaugrande & Dressler 81]. This section will therefore start with a summary of the principles of text organisation taken from [deBeaugrande & Dressler 81]. This will be followed by a discussion of specific principles which may apply to particular types of discourse and a summary of the criteria used for the EDGE model.

4.1.1 Standards of Textuality

Beaugrande and Dressler suggest seven standards of *textuality* which may be used to guide the construction of texts if they are to be communicative. These standards will be discussed in turn, along with a brief discussion of how the text generation systems discussed in chapter 2 meet these criteria, and how the principles relate to explanatory discourse. The first two (cohesion and coherence) concern syntactic and semantic relations between text segments, independent of the hearer. The next three (intentionality, acceptability and informativity) refer to the goals of the speaker and hearer and to the hearer's ability to assimilate the information. Finally, situationality refers to other aspects of the discourse situation as they affect the text, while intertextuality considers how text is influenced by other related texts.

4.1.1.1 Cohesion: Surface Level Ties

The first principle relates to the surface organisation of text. Cohesive relationships are surface level ties between elements in a text, and have been discussed in detail by Halliday and Hasan [Halliday & Hasan 76], for example. Cohesive relations include the use of pronouns, lexical and syntactic repetition, ellipsis and the use of appropriate conjunctions.

Cohesive relations are clearly important, and may increase the effectiveness of all text types. They may be used for economy of presentation, to ease processing load, or to create special effects (such as in poetry). While recognising its importance, this thesis does not address such aspects of surface expression, concentrating on the structure of the overall discourse. However, if text is to be readable, certain aspects cannot be ignored. The most important of these is perhaps the use of pronouns. For example, the following examples of two consecutive sentences seem very unnatural:

- (1) The light detector unit is a sort of potential divider circuit.
- (2) The light detector unit's function is to provide an output voltage which depends on the input light intensity.
- (1) Today we are going to make chocolate chip cookies.
- (2) Do you know how to make chocolate chip cookies?

The repetition of noun phrase and verb phrase respectively not only makes for uneconomical presentation, but destroys the structure of the discourse, encouraging the hearer to believe that the second sentence begins a new topic. The appropriate use of pronouns will therefore be considered essential to the generation of communicative text and should be used even in a simplistic treatment of sentence level generation.

Several text generation systems have considered how pronouns and referring expressions should be selected, for example [Dale 89]. A simple basis for pronoun selection may be adapted from [Dale 86].

4.1.1.2 Coherence: Conceptual Relations

The term 'coherence' will here be used to refer to the relations between the concepts or ideas in the text. A coherent text is one where the concepts and relations underlying a surface text are accessible to the reader. Much of the coherence in a text may be implicit, and only accessible by inference using commonsense and specific knowledge¹. For example, the utterances:

Let's go for a pizza. Muck's gone down. It's the third time this week.

would not seem coherent to someone who didn't realise that the speaker was the user of a computer called Muck (named after the Scottish island), who had nothing to do while his computer was down. Of course, if the background

¹Carlson emphasises such 'implicit moves' in his dialogue game based theory of discourse structure [Carlson 83].

information was known then including it would appear verbose and unnatural. Indeed, some types of discourse (such as jokes) seem to require such inferences.

So, whether they are explicit or not, there should be relations between the concepts underlying a text. These may describe links between the objects and actions mentioned in the text, or between the ideas in whole sections of text. Both these types of constraint have been used in recent text generation systems.

Focus rules have been used to constrain how the objects in focus in the text can change as the discourse progresses. These may capture both how focus is influenced by the structure of the discourse (returning to previously mentioned items) and how focus can move to new, unmentioned items (such as an attribute of an object in focus). McCoy and Cheng's work on focus trees perhaps makes these focus movement rules clearest [McCoy & Cheng 88], though McKeown also used focus rules to constrain generation [McKeown 85].

Mann and Thomson's 'Rhetorical Structure Theory', introduced in section 2.1.2, is based on relations between ideas in whole sections of text [Mann & Thompson 87]. A restricted set of relations (such as 'compare-contrast') may be applied hierarchically to describe text structure. These relations form the basis of Hovy's text planner [Hovy 88a] and have been used recently in conjunction with focus trees [Hovy & McCoy 89].

These different types of coherence may be illustrated using the 'pizza' example above. A possible RST structure would be to view the third utterance as an *elaboration* of the second, and to view the whole text as containing an implicit *persuade* relationship containing some action (sentence 1) and some reason (sentences 2 and 3). Focus movement between the first and second sentences is very complex. However, in the second and third sentences the focus shifts from the computer (Muck) to the event of it 'going down' (a potential focus in sentence 2).

In explanatory discourse (such as McKeown's descriptive texts), few of the RST relations apply, and the resulting text structure will generally consist of a number of 'elaboration' relations. Indeed, Moore and Swartout's planner con-

sists primarily of operators which are specific versions of the elaboration relation [Moore & Swartout 88b]. However, coherence appears to be maintained in several ways. First, general focus movement rules will apply, as described by McCoy and Cheng. Second, normal orderings are used between events or situations (such as temporal, causal, spatial or justificatory). Finally, there are clear sub-goal and prerequisite relations between parts of the explanation, derived from the intentions of the speaker and sub-skill relations between concepts being taught or explained. This last point is developed below.

4.1.1.3 Intentionality: The Speaker's Goal

Text structure is clearly also influenced by the goals of the speaker. Normally, text will only succeed in its communicative objectives when that intention is recognised. For example, the utterance:

The door's open.

will only succeed if the hearer recognises the speaker's intention to get the hearer to shut the door. Such indirect speech acts may be interpreted through knowledge of the likely goals of the speaker [Searle 69] and principles of communication [Grice 75].

However, intentions in the sort of explanatory discourse being considered are very simple. The speaker intends that the hearer understands some complex piece of knowledge. So the more interesting issue is the structure of the discourse which results from this intention. The process of 'understanding' may be complex, requiring a mapping between the new knowledge and the hearer's existing knowledge, so the resulting discourse will depend on both of these.

The intentional structure of the text appears to be based on the sub-skill and prerequisite relations between the concepts being taught². Given a goal to teach

²An alternative view is to consider the intentional structure as arising from attempts

a particular skill, this may involve teaching prerequisite skills, and decomposing the main skill into sub-skills. This approach has been used in Intelligent Tutoring Systems to control the curriculum (as described in section 2.1.2), yet has not been applied to generating explanatory text.

Grosz and Sidner's model of discourse structure (discussed in chapter 2) emphasises the intentions of the speaker, and how these relate to the linguistic and attentional structure [Grosz & Sidner 85]. They fail to describe in detail how the intentions of a speaker may result in particular text structures (unlike Mann and Thomson). However, given a plan-based approach to text structure, their theory may be used to determine the effects on surface linguistic expressions and focus shifts. A model of explanatory discourse can therefore be based on this theory, combined with knowledge of how sub-skill and prerequisite relations may be used in planning explanatory content.

4.1.1.4 Acceptability: The Hearer's Goals and Attitudes

The acceptability principle relates to the hearer's attitude to the text. If textual communication is to succeed, then the hearer must recognise the text as relevant to her goals.

If the hearer's goal is to acquire the knowledge which the speaker is conveying, then things are simple. But in general the hearer will have some less direct goal for seeking an explanation. For example, an explanation of how an engine worked might be requested because the hearer wanted to pass an exam, mend her car, run her car more efficiently, or out of curiosity. Sometimes the goal might be even less direct — such as to impress or to begin a conversation with the explainer.

to transform or develop the hearer's mental model of the world. However, where the explanations are extending rather than transforming the model, the simpler view based on prerequisite skills that link with existing knowledge is adequate.

The content of the explanation should ideally depend on this higher level goal of the hearer.

Some recent work on text generation (discussed in chapter 2) has been concerned with these issues. For example Hovy considers how text may be generated which depends on the attitudes of the hearers [Hovy 88b] while McKeown, Wish and Mathews, and van Beek are all concerned with how expert systems explanations should depend on the hearer's goals [McKeown *et al* 85, vanBeek 86].

In this thesis it will be assumed that the hearer's goal is simply to assimilate the new information! This is a convenient idealisation rather than a sound basis for explanation, so should be seen only as a useful starting point.

4.1.1.5 Informativity: The Hearer's Knowledge

Text should be sufficiently informative, without overloading the hearer. Even if a piece of text addresses the goals and interests of the hearer, it may fail to be communicative if too much or too little new information is given, if prerequisite information is missing, or if the 'model' used in the explanation is inconsistent with the hearer's.

Some suggestions have been made in Intelligent Tutoring Systems concerning the amount of new information that should be given. At one extreme, Van Lehn shows that 'one disjunct' of a procedure is typically taught in a single lesson [VanLehn 83]. Wenger attempts to generalise this to smaller units of teaching [Wenger 87, pg 336], and Draper suggests that an explanation should include one new bit of information and lots of links into existing knowledge [Draper 87].

In general, if the text is to be informative (recognisably coherent to the hearer), then concepts in the text should not be introduced that cannot be linked in with the hearer's existing knowledge. For example, the sentence "The light detector unit is a potential divider circuit" is not informative if the hearer is unfamiliar with the potential divider circuit. Generating informative text,

then, involves defining 'prerequisite' relations between concepts such that if all the prerequisites are understood then the hearer should be able to assimilate the new information.

Of course, it may be insufficient to simply supply prerequisite information and avoid introducing too much new material. The hearer's knowledge (and mental model [Gentner & Stevens 83]) may mean that very different strategies should be selected to explain something (for example, using different models of electricity [White & Frederiksen 85], or using an analogy to describe something).

Much of the early work on explanation and text generation used the simple notion of not telling the hearer things she already knows [Mann & Moore 81, Wallis & Shortliffe 84]. However, there has not been a systematic treatment of informativity.

4.1.1.6 Situationality: The Discourse Context

Beaugrande's sixth standard of textuality seems to subsume many of the others, and concerns the factors that make a text relevant to its situation of occurrence. Two aspects of the situation will be considered. First, the immediately surrounding discourse will influence what will be said. For example, the hearer's responses to past utterances will influence future utterances. Moore and Swartout, for example, show how answers to follow-up questions can take into account the previous explanation given [Moore & Swartout 88b].

Second, the wider discourse situation will influence the text. Halliday uses the term *discourse register* to denote the set of factors which affect the discourse context [Halliday 73]. These are the *field* or subject matter of the discourse, the *mode* of interaction (e.g., fact to face verbal), and the *tenor* of the discourse, which includes the roles and attitudes of the participants. Other factors might include the time available to speak and the current activities of the participants. For some types of discourse (such as task dialogues) the situation has a direct effect on the discourse — the expert's monitoring of the situation will influence

what she says and when she says it. For other types of discourse, situational effects are more indirect. In a classroom, the social roles of the participants will have a strong influence on what they say (and may be reinforced by the discourse unless deliberate attempts are made to break out of the pattern). In a courtroom the conventions are even stronger.

In explanatory discourse it should be possible to use both the immediate and general discourse context to influence the text. The general discourse context might have a particularly strong influence on the presentation, given a particular discourse goal.

4.1.1.7 Intertextuality: Text Types

Interpreting a text may require knowledge of other texts. This is referred to as intertextuality. Some types of discourse (such as parodies and reviews) rely directly on an understanding of some other text. However, the term 'intertextuality' may also refer to the existence of particular text types. For example, conversations, scientific papers and poems all have particular rules and stylistic conventions. While these may be related to the goals and social roles relevant to that discourse type, they are more likely to be abstracted by speakers from their knowledge of other texts of that type. In general, generating a text will combine the use of these rules and structures known to be relevant for that particular discourse type with planning to meet new goals, situations and hearers [Clark & Clark 77, pg 236]. Unusual situations and difficult goals will require more active planning, while common situations may require little thought, using pre-existing schemas. As writers become more expert, certain types of discourse become 'compiled' and can be followed more or less automatically.

A distinction that should be made is between the structure of the propositional content (which the hearer is supposed to be assimilating) and the structure of the text as a whole. This distinction is made for example by Kieras, who uses *content schemas* to represent the stereotypical patterns used for arranging the

content (facts) in different domains, and *text grammars* to represent more domain independent configurations of textual elements [Kieras 85]. For example, there may be a content schema which is used to describe a class of physical objects, while a text grammar might state that in explaining something you should first state the topic, expand on it, then present a conclusion. These two types of textual organisation are worth analysing in more detail.

Content Schemas

Content schemas, or stereotypical organisations of propositional content, have been observed by many researchers (e.g., [Rumelhart 75, Linde 77, Weiner 80]), and provides the basis for the text generation systems described in [McKeown 85, Paris 87]. However, schemas may often be explained by more general constraints. For example, Wilensky argues that story structure can be explained by considering the plans and goals of the participants in the story, and that interpretation involves reasoning about these plans and stereotypical action sequences [Black & Wilensky 79]. Mandler and Johnson, in their response to Wilensky, argue that both world knowledge and specifiable structures (story grammars) are used in interpretation and recall [Mandler & Johnson 80]. Similarly, in chapter 3 it was argued that schemas capturing common patterns of explanation could be partially explained by 'prerequisite' constraints between concepts in the explanation, while various sorts of coherence constraints may further 'explain' the schemas.

It appears then that content schemas (representing typical organisations of text in particular domains) are based primarily on the sorts of principles discussed above. Main organisational principles include the organisation of domain knowledge; prerequisite relations between concepts; coherence relations between concepts; and the goal structure of the discourse. Writers may use stereotypical schemas when generating common (or highly conventionalised) text types, but

may always have recourse to more general knowledge about the domain and the hearer and may plan text to achieve particular effects.

Text Grammars

Text consists of much more than an arrangement of facts or propositions. Summaries, introductions, headers and different types of *meta-comments* are used to aid the hearer in their understanding. This is especially true in educational discourse, where such comments (and others which serve to 'oil' the communication channels) may represent the majority of utterances [Stubbs 83, Sinclair & Coulthard 75]³.

Let's consider the structure of this paragraph here. First, it begins with a meta-comment describing what is about to be discussed. Second, the words 'first', 'second' and 'finally' are used to indicate the structure of the content. Finally, the paragraph ends with a conclusion, summing up what has been said. So, we can see in this paragraph that even simple text sections use meta-comments, clue words and summaries to aid understanding.

Textual elements such as these, which may not convey any new propositional content, serve to make it easier for the reader to understand the text. Meta-comments and headlines may serve as 'advance organisers', directing the reader to the relevant knowledge. Clue words (such as 'OK', 'Now', 'Anyway') serve to indicate the structure of the text, which may in turn indicate the topic

³In this discussion we are only concerned with meta-comments which serve to structure the discourse, prepare the user for what follows or refer back to what has been said. Carbonell and Stubbs both discuss a wider range of meta-linguistic acts, including correcting previous utterances [Carbonell 82] and controlling the amount of speech [Stubbs 76].

structure⁴. Summaries both indicate the close of a topic, and again help organise the knowledge.

One existing text generation system which uses such expressions is Zuckerman and Pearl's work, described in chapter 2 [Zuckerman & Pearl 86]. However, they don't consider how the use of such expressions depends on the type of discourse. For example more summaries may be used in technical papers and more meta-comments in teaching discourse. It should be possible to define some sort of 'text grammar' to describe how these comments are used in particular types of discourse, independent of the actual content. For explanatory discourse these may be used to describe how discourse markers and meta-comments relate to the topic structure of the text (combining ideas from [Sinclair & Coulthard 75] and [Grosz & Sidner 85]).

Although standard 'text types' (both text grammars and content schemas) are undoubtedly useful in generation, they are generally inflexible, not generalising well to other domains or situations or allowing appropriate responses in a dynamically changing situation. Where possible, we should seek to explain these stereotypical structures. However, examining text structure in this way has led to an important principle — that text should be planned at (at least) two levels. The first of these may be concerned with the organisation of a particular type of discourse independent of content, while the second may be concerned with content organisation⁵.

⁴The various uses of different sorts of clue words has been discussed by Reichman [Reichman 81].

⁵Note that distinguishing between these different levels of textual organisation helps explain the difference between RST relations such as 'persuade' or 'motivate', relations such as 'cause' or 'sequence' and schemas such as a 'social letter' schema [Hovy 89]. In a multi-level system it makes sense to talk about the structure of a '*social letter* to

4.1.1.8 Conclusion

There are clearly many principles which influence the structure and content of multi-sentence text. Existing text generation systems may make one or two of these principles explicit, while other principles are implicit in schema-like orderings of types of description. Progress will be made when more of these principles are made explicit, allowing far more flexible generation. The EDGE system attempts to make several of these principles explicit, though emphasising the *informativity* principle.

4.1.2 Domain Dependent Criteria

The principles discussed above apply to all texts. Yet there are other criteria which apply to particular domains or types of discourse. For example, Mayer gives a list of fourteen criteria for structuring technical explanations [Mayer 85]. The majority of these criteria are suggestions about how to highlight important topics so they will be learnt by the hearer, while the other criteria are based on encouraging the hearer to actively think about the explanation by using concrete models, analogies and questions.

When interaction is possible, explicit questions should be asked (and the answers analysed) and interruptions allowed. Furthermore, by coordinating text with a graphical display the explanation should be made more concrete. These two principles will be developed further in chapters 5 and 7.

persuade someone that some *causal relation* exists'. This may be similar to McCoy's use of both communicative roles and content in response schemas.

4.1.3 Summary

The above discussion has been fairly general, introducing a number of principles for generating effective communicative text. This section will summarise the actual criteria which the EDGE model will satisfy. These are necessarily only a selection of possible criteria, and perhaps merely define the scope of this investigation. For example, the hearer's goal is not considered in the criteria.

Cohesion: The system should attempt to use pronouns appropriately, using a simple model of discourse focus.

Coherence: Two types of coherence should be considered:

- There should be some relation between items in focus in successive sentences.
- Standard ordering conventions (e.g., causal or temporal) should be used where appropriate.

Intentionality: The discourse should be derived from an overall goal of the speaker (and not from arbitrary schemas) and this *intentional structure* should be explicitly represented. In explanatory discourse the primary intention will always be to convey some complex concept.

Informativity: Text should be planned so that it may be easily understood by the hearer. This includes:

- Selecting models and strategies for explaining which fit in best with the hearer's knowledge (for example, using an analogy in describing).
- Including prerequisite information as required, so that each new concept may be linked in with one that the hearer already has.
- Using clue words, meta-comments and summaries where appropriate, to help the user recognise the topic structure and assimilate the new information.

Situationality: It should be possible to vary the text depending on both the immediate and general discourse context. This could include:

- Using the immediate discourse context to influence what is said.
- Using the general discourse context to influence how it is said (e.g., politely or patronisingly; using meta-comments and questions or without).

Text Types: Text structure should be planned at two levels:

- **Content Schemas:** It should be possible to define standard organisations of text content used to achieve particular effects. However, these should not be used where there are more general constraints which could be applied.
- **Text Grammars:** It should also be possible to define how a particular text type should be presented, including the use of meta-comments, for example.

Interaction and Graphical Actions: For certain domains and types of discourse it should be possible to allow interaction and coordinated text and graphics.

4.2 The EDGE Text Planning System

This section will describe the EDGE explanatory text planning system⁶. A distinction is made between planning the text content, and planning how to present it to the hearer. The former is based on a set of *content* plans, while the latter is based on a set of *discourse* plans. The text content will depend on the domain knowledge of the hearer, and will be planned so that it may be assimilated by the hearer, while satisfying the main explanatory goal. The way this is presented to the hearer may depend on the discourse situation, yet is largely independent

⁶Note that though the complete EDGE system can be used for planning interactive discourse, the basic framework can also be used for planning uninterrupted multi-sentence text. It is this part of the model which is described here.

of the domain, so will be specified separately. This distinction corresponds to Kieras's distinction between text grammars and content schemas. It also corresponds to the common distinction made between 'what to say' and 'how to say it', though this 'how to say it' includes high level discourse phenomena.

It is worth mentioning the history of this representation. In an initial version, a 'schema' was written which could be used to generate explanations in the chosen domain which depended on the hearer's knowledge. However, in order to generate coherent text, meta-comments and clue words had to be included in this schema. At the same time, a formal basis was being developed for the control of the dialogue in the explanation (described in the next chapter). The discourse theory on which this was based included a description of the use of meta-comments and clue words. By transferring them into this part of the model, the schemas became a simple representation of content organisation.

The second stage of development was to note that these schemas related to the conceptual structure of the domain. A simple change in the representation and the content schemas became a set of plan operators for teaching abstract concepts. These described prerequisite and sub-skill relations between concepts. This also enabled an improved representation of the user's domain knowledge. This could now be described at many levels of detail from general concepts or skills to the smallest mislearnable sub-skill.

The following sections will describe the EDGE discourse planner, and how it is used to plan the content and the presentation of an explanation. The system is fully implemented in Interlisp-D on a Xerox 1186 machine⁷, and the example plan definitions which follow are taken from the implemented system. However, it should be possible to understand these without a familiarity with Lisp. Most of the examples will be taken from an implementation used to generate tutorial

⁷These machines were donated by the Rank Xerox Corporation to the University of Edinburgh under the Rank Xerox University Grants Program.

explanations of how simple circuits work, but the section will conclude with a discussion of how the basic formalism may be (and has been) applied to other domains and types of discourse.

4.2.1 The EDGE Discourse Planner

The EDGE discourse planner is basically a simple skeletal planner [Friedland & Iwasaki 85] where high level goals may be either decomposed into more detailed sub-plans, or specialised into instances of the plan which apply in particular situations⁸. *Constraints* on the plans determine when a particular plan applies, and are generally used to select a particular specialisation of a plan (such as to choose whether to describe using an analogy).

4.2.1.1 Planning Operators

The basic skeletal plan representation is extended to allow *preconditions* to be defined. This is a type of subplan which is only executed when the user does not know the associated skill or concept, and only applies to content plans. In the content plans, the plan operators refer to the abstract concepts which are being taught. The subgoals can therefore be interpreted either as a method for teaching the skill represented by the parent plan, or sub-skills which, if understood, would mean that the parent skill should be understood. There is therefore a fairly subtle distinction between subgoals and preconditions — subgoals represent parts of a skill or concept, while preconditions refer to other concepts which

⁸Skeletal planners contrast with conventional planners in that a plan is constructed from pre-built plan fragments, rather than built from scratch from planning operators with preconditions. They are less powerful, but more efficient where there are standard ways to achieve goals. The formalism is actually closer to, say, Paris's recursive application of schemas than to a conventional AI planner.

must be understood if the main concept is to be successfully learnt. With limited understanding of how people learn new skills or concepts it may often be unclear how to structure content plans.

Discourse plans have the same structure as content plans, but obviously don't have the same association with skills or concepts. Each planning operator (content or discourse) may have:

Name: For content plans this describes the skill being taught in the plan⁹. For other sorts of plans it is simply used in plan selection.

Arguments: Arguments will generally be domain objects in content plans, and content plans or propositions in discourse plans.

Preconditions: A list of prerequisite skills for content plans.

Constraints: Applicability conditions, for selecting alternative instances of a plan based on the discourse context and user model.

Subgoals: Subgoals which should all be executed, including sub-plans.

Template: Accessed by the discourse planner in giving meta-comments and currently only used in content plans.

A simple example plan illustrating many of these features is given below:

⁹As each plan teaches a single skill the name of content plans effectively defines the 'effect' of this plan — the representation therefore becomes similar to conventional planners reasoning about effects where those plans have a single effect.

```
(defplan structure (device)
  :preconditions ((what-sort-of-device (device)))
  :subgoals ((teach components (device)))
  :template ("what" (deviceref device)
             "is like (its structure and components)"))
```

In this example the *name* of the plan is **structure**, and it has one *argument* — **device**. It has a single precondition and subgoal and a template describing the concept being taught. This example does not have any constraints on its applicability. Other plans may have no preconditions.

The operators are similar to the planning operators used by Litman and by Carberry in recognising user goals in discourse understanding [Litman 85, Carberry 83].

4.2.1.2 Subgoals

Subgoals of plans may include sub-plans, conditional calls, iterative calls over a set of objects and Lisp calls. Lisp calls represent the terminal items of a plan. They are used to execute speech acts or graphical actions. However, as they generalise the representation they should not be used otherwise¹⁰.

4.2.1.3 Planning Algorithm

The same planning algorithm is used for planning both the content and the presentation. A simple goal stack is used to control the planning. The basic algorithm is then to take a goal off the stack and, depending on its type, either

¹⁰An exception to this rule is where a complex reasoning process is needed to obtain the steps of an explanation (such as a proof or causal behaviour of a system). Then the reasoner may be called directly and pass control back to the planner when the reasoning steps have been found.

execute it or replace it with new goals. If the goal is a plan the new goals will be the subgoals of the plan. If there are any preconditions, these are replaced with *conditional* goals which will only be called if the associated concept is not understood.

Planning proceeds incrementally — subgoals are expanded and put on a stack until the next item is a speech act or graphical action. This is executed before the planning continues. The planning process is currently applied deterministically — once a planning decision is made it cannot be retracted. However, it would be possible to allow a limited amount of ‘backtracking’ within the incremental planning framework

The basic algorithm is summarised below:

- A goal is taken off the stack.
- Depending on its type then:
 - If it is a Lisp call it is immediately executed.
 - If it is a conditional goal then if the condition is true the body of the goal is put on the stack.
 - If it is an iterative call then each goal is put on the stack.
 - If it is a plan then an instance of that plan which satisfies all its constraints is found and the subgoals of that plan put on the stack (replacing preconditions with conditional plans).

Whenever goals are put on the stack their arguments are evaluated in the current context (using the actual arguments of the parent plan and any current global variables). Thus, in the example plan above the argument ‘device’ would be replaced by the actual device being explained. In a more complex example an argument, such as ‘(getslot device ’structure)’, may be evaluated and replaced by the structural description of the device being explained¹¹.

¹¹Note that in the representation used the arguments to be evaluated are given in a

As well as being added to the stack, goals are added to a global representation of the plan. This is a simple discourse model where the subgoal and prerequisite links between goals represent the intentional structure of the discourse, and the evaluated arguments represent a simple focus model. The use of the discourse model will be discussed further in chapter 5, but a simplified fragment of an example model is given below:

```
(teach how-it-works (light-detector))
  (teach causal-behaviour (light-detector))
    (teach particular-causal-behaviour                ****
      (light-detector (light-intensity high)))
    (teach particular-causal-behaviour
      (light-detector (light-intensity low)))
  (teach what-it-does (light-detector))
```

This example only illustrates the plan of the content, and not of the presentation (the full example is given in figure 5-3). The stars indicate the current stage in planning. The goals beyond this point will be on the goal stack and will be expanded as planning proceeds.

The other evolving representation is the user model. This will be continually updated (see chapter 6). Planning decisions may depend on the user model and on the discourse model (plan). As the planning proceeds incrementally these may take into account unexpected events changing the discourse structure and revealing new knowledge about the user.

The following sections will show how this representation is used to plan content and presentation of text.

list. So the (hypothetical) plan '(teach what-it-does (inputs device))' would have the two arguments 'device' and 'inputs' evaluated, while the plan '(teach what-it-does ((inputs device)))' would apply the function 'inputs' to the evaluated argument 'device' to find the value of its single argument.

4.2.2 Planning Text Content

An explanation begins with a goal to teach or describe some complex task, skill or concept. Planning operators are used to decompose that skill into primitive sub-skills, and to fill in prerequisite information which is required by a particular user. The set of all the planning operators defines the range of all possible explanations, and this should correspond to content schemas or grammars derived from empirical analysis of explanations in that domain.

An example operator (used to teach how a physical system worked) might be:

```
(defplan how-it-works (device)
  :constraints ((greaterp expertise-level novice-level))
  :preconditions ((structure (device)))
  :subgoals ((teach causal-behaviour (device))
             (teach what-it-does (device)))
  :template ("how the " (getslot device 'name) " works "))
```

This operator should be interpreted as follows. This is a plan to teach how a given device works applicable when the user is not a novice in the domain. The user must first be familiar with the structure of the device. To teach how something works, first give a causal description of its behaviour, then describe what it therefore does.

Note that the basic plan representation could easily be used to combine schemas such as those used by Paris [Paris 87]. Two basic schemas would be defined for describing a component of a device, with the constraints that the user should be an expert or novice about the particular component.

4.2.2.1 Example Content Planning

The set of operators may be used to generate the complete explanation content. Initially, an operator call such as '(teach how-it-works (light-detector unit))' is put on the stack. The 'planner' proceeds by repeatedly taking an item off the stack, and applying the algorithm given above. The following example illustrates

this process — more detailed examples of the complete system are given in chapter 7.

The following operators describe how to teach what a device does for different inputs¹².

```
(defplan what-it-does (device)
  :subgoals ((foreach input in (getslot device 'inputs)
    (teach in/out (device input
      (get-output device input))))))

(defplan in/out (device input output)
  :subgoals ((interaction teaching.exchange
    ((list 'describe-event
      device input output)))))
```

Suppose the initial goal stack is:

```
((teach what-it-does (LDR)))
```

The planner now looks for an operator which could be used to teach that concept. The only one (having no constraints on applicability) is the one above. As there are no preconditions, the subgoals are expanded. The possible inputs to an 'LDR' are a high or low light intensity which give a 'lowish' or 'highish' output resistance, so the goal stack becomes:

```
((teach in/out (LDR (light-intensity high) (resistance lowish)))
 (teach in/out (LDR (light-Intensity low) (resistance highish))))
```

The planner takes the first item off the goal stack, and looks for a suitable operator. Using the second operator given, the subgoals are expanded and the goal stack becomes:

¹²These operators are slightly simplified versions of the actual ones used, given in appendix B. The teaching exchanges are described later in this chapter, and fully in the next chapter. Their function is to convey the given proposition to the user.

```
((interaction teaching.exchange
  (describe-event LDR (light-intensity high)
    (resistance lowish)))
  (teach in/out (LDR (light-intensity low) (resistance highish))))
```

Again, the planner takes the first item from the goal stack. As it is a proposition to express, control passes to the discourse planner (discussed in the next section)¹³ resulting in the following sentence being expressed.

“When the input light intensity of the light dependent resistor is high, the output resistance will be low”

The remaining item on the stack is expanded, and the second utterance generated:

“When it is low, the output resistance will be low”.

4.2.3 Planning the Presentation

Given some content, the system must decide how to convey it to the user. This may involve adding clue words, meta-comments and possibly summaries and introductions. It must also involve deciding how to convey each proposition. Zuckerman and Pearl, for example, concentrate on adding meta-comments of different kinds to improve the presentation of a mathematical explanation [Zuckerman & Pearl 86]. Van Dijk describes how people may generate news articles, given the *thematic structure* of the item. This involves selecting the most important items for the headline, and traversing the thematic structure to extract the lead and the opening paragraphs [vanDijk 85].

¹³Propositions are always passed to the discourse planner for expression. In the current system the only directly executable actions in the content plans are graphical actions. It is debatable whether even they belong in the content plans.

4.2.3.1 Operators and Discourse Structure

In this thesis, the same structures are used in planning the presentation and controlling the dialogue. They will therefore be described in greater detail in the next chapter. This section will therefore describe these aspects only briefly, in order to complete the description of text production.

The same sorts of planning operators are used in planning the presentation as are used in planning the content. The approach is adapted from the structures suggested in work on discourse analysis (especially classroom discourse [Sinclair & Coulthard 75]), introduced in chapters 2 and 3. In this theory there are five levels of structure, each level constraining the possible sub-types in the level below. The levels used in generating explanatory discourse are:

Transactions — This is the top level, including such things as describing or explaining some topic. In the adapted theory used, sub-transactions are also allowed for sub-topics and interruptions.

Exchanges — These may be things like question-answer-response sequences, inform-acknowledge, or boundary exchanges at the beginning of a topic.

Moves — The individual moves within the exchanges.

Acts — The linguistic acts within the move.

Sets of planning operators are defined for each type of interaction. In this chapter we will only be concerned with those which affect the presentation independent of any dialogue.

The following are some of the top level interaction definitions.

```

(definteraction informing.transaction (teaching-goal)
  :subgoals ((interaction boundary.exchange
                  (teaching-goal 'open))
              (interaction teaching.exchanges
                  (teaching-goal))
              (interaction boundary.exchange
                  (teaching-goal 'close))))

(definteraction teaching.exchanges (teaching-goal)
  :subgoals (teaching-goal))

(definteraction boundary.exchange (teaching-goal type)
  :constraints ((eq type 'open))
  :subgoals ((interaction framing.move (type))
              (interaction focussing.move
                  (teaching-goal type))))

```

These plan operators should be interpreted as follows. The first is used to structure a transaction on a topic. This starts with an opening boundary exchange, which is followed by a sequence of teaching exchanges on the topic (or, in uninterrupted text, a sequence of utterances on the topic) and a closing boundary exchange. The teaching exchanges plan operator passes control to the content planner — the teaching goal is evaluated and put on the goal stack. As content planning proceeds this teaching goal will have its subgoals or plans filled in and will become a complete content plan. It will be the content plan which will determine the content of individual teaching exchanges. The final plan operator illustrated — the boundary exchange — shows that an opening boundary exchange consists of a frame move (a discourse marker) and a focussing move (a meta-comment).

In planning discourse, control may switch freely between the discourse planner and the content planner. For example, if some complex content is planned, another informing transaction could be initiated from within the content plans. So, the resultant plan includes both discourse plans with content plans as their subgoals and content plans with discourse plans as subgoals. Discourse plans may be viewed as a type of meta-plan (cf. [Litman 85]) as they may take content plans as their arguments. However, the operations which may be done to

the content plans are currently restricted to accessing and calling them (putting them on the goal stack). A more general theory would have to consider how content plans could be transformed to improve presentation.

4.2.3.2 Example

So, a typical top level call to be put on the plan (goal) stack would be:

```
(interaction informing.transaction
  ((teach how-it-works (light-detector))))
```

The planning process would expand this, so the stack would next consist of:

```
(interaction boundary.exchange
  ((teach how-it-works (light-detector)) 'open))
(interaction teaching.exchanges
  ((teach how-it-works (light-detector)))
(interaction boundary.exchange
  ((teach how-it-works (light-detector)) 'close))
```

Now, an opening boundary exchange consists of a frame and a focussing move¹⁴. A frame move will be realised as a clue word such as 'Now', 'OK' or 'Right', while a focussing move may consist of one or more meta-comments. In order to generate meta-comments, the system accesses the template given for the teaching plan operator. So, using the definition of the 'how-it-works' operator given in the previous section, the following utterance will be printed:

Right, We're going to be explaining how the light detector circuit works.

¹⁴As noted in chapter 3, closing boundary exchanges in one to one explanatory discourse are designed to check that the novice is ready to leave a topic and to give the opportunity for follow-up questions, and so have a different form — this will be discussed in the next chapter.

If the system reasons that the topic is especially hard, or especially important, further appropriate meta-comments may be generated.

Now the second item on the plan stack given above will be expanded, using the second operator given, resulting in the following plan stack:

```
((teach how-it-works ((light-detector)))  
  (interaction boundary.exchange  
    ((teach how-it-works ((light-detector))) 'close)))
```

Control will now pass to planning the text content, as described in the last section. It would pass back to the discourse level either when there is a proposition to convey, or when a new topic is introduced and a new transaction is initiated.

4.2.3.3 Presenting propositions

The preceding example illustrated how meta-comments and some clue words are introduced into the discourse. This satisfies our criteria that such constructs must be used to help the user assimilate the new information and understand the topic structure of the discourse. The other aspect of presentation considered is how to convey the propositions in the content plan. A simple template-based approach is used. However, some features of this approach are worth discussing.

Given a proposition to convey, the system may either ask a question, or present the information. In this section only the latter will be considered. So, suppose that the system has to convey the following proposition:

```
(describe-event LDR (light-intensity high) (resistance lowish))
```

The system then accesses the appropriate 'describe-event' template. There may be a set of templates for a particular proposition, with constraints on their applicability. Currently alternative templates are available depending on the

number of inputs to an event, for example.¹⁵ For the above example, the following template is accessed:

```
("When " (getref (gettype input) (list "the input " (gettype input)))  
"is " (getval input) ", the output " (gettype output) " is "  
(getval output) ".")
```

This rather complex template includes functions to obtain English versions of internal values, and select whether the pronoun 'it' should be used (getref). This latter procedure uses the stored context based on the arguments in the last proposition expressed to decide whether or not to use the full expression.

4.2.3.4 Selecting Pronouns

Pronouns may be used both for domain objects and domain plans, with the arguments of plans giving the current context. These arguments may be domain plans (arguments of discourse plans) or domain objects (arguments of domain-content plans). Thus pronouns may be selected for the following sorts of cases:

- (1) Now, I'm going to explain what a light dependent resistor does. It's quite important that you understand *that*.
- (2) A light detector unit is a type of potential divider circuit. *Its* function is to give an output voltage which depends on the input light intensity.
- (3) Whenever the input light intensity is high, the output resistance is quite low. Whenever *it* is low, the output resistance is quite high.

The current implementation uses the context of the previous sentence in deciding whether or not to use a pronoun. The text template may have its

¹⁵In an earlier version of the system simpler templates were available for propositions believed understood by the user, but in practice it was hard to simplify templates without assuming the context in which they would be used. This illustrates one of the problems of using templates which are supposed to apply in a range of contexts without an explicit treatment of ellipsis, for example.

principal focus (or *centre*) indicated (by inserting a 'getref' function) to show that if that item is in the current context then a pronoun should be used. This works satisfactorily in the majority of cases. However, there are cases where a pronoun is used where it shouldn't be, such as when a new sub-topic begins where its principal focus is in the context of the last sentence of the previous topic, or where the principle focus changes between utterances. In these cases a model based on Grosz and Sidner's discourse theory [Grosz & Sidner 85] would improve pronoun selection — this would be a simple extension given the existence of a discourse model consistent with this theory¹⁶.

The very simple model for surface generation and pronoun usage obviously has many limitations. However, it does allow some cohesion to be maintained by using pronouns and provides a good basis for further development.

4.2.4 Varying the Domain or Situation

So far the discussion has been based on explanations of how simple physical systems function. Yet the basic approach will generalise to any domain or type of explanation where there is a complex skill or concept to be taught. This skill may normally be defined in terms of sub-skills and prerequisite skills, and content planning operators defined. Depending on the roles of the participants, different discourse plans may be used¹⁷.

¹⁶Other problems in pronoun selection include cases where there are multiple possible referents in the last sentence (requiring semantic knowledge in the pronominalisation process), and where the sense of the item differs from the item in the previous sentence while the reference is the same — it is the referent that is used in the pronominalisation procedure in the EDGE system.

¹⁷It should also be possible to generalise the basic approach further, such as to narratives, justifications or persuasive arguments. These too have a hierarchical structure

One large class of such explanations involves descriptions of how to do some task. Such tasks might include doing some complex mathematical operation, constructing or mending a device or cooking a meal. In order to investigate this class of explanations, and assess the generality of the basic approach, the explanation generator described above was modified to generate explanations of how to make chocolate chip cookies.

4.2.4.1 Changing the Type of Domain: Task Descriptions

Task dialogues have a simple structure which has been investigated by several researchers (e.g., [Grosz 77]). The structure of the discourse corresponds closely to the structure of the task. Uninterrupted task descriptions similarly follow the task structure closely. The level of detail (i.e. whether sub-tasks are described) depends on whether the reader knows how to execute these sub-tasks. Prerequisite tasks should be mentioned, if they have not already been satisfied.

The planning operators for tasks are therefore at least superficially similar to the ones used in the electronics explanations. For example, two task descriptions used in the 'cookie explainer' are:

```
(defplan make-cookies
  :preconditions ((get-ingredients))
  :subgoals ((plan make-cookie-mixture)
             (plan roll-out-cookies)
             (plan cook-cookies))
  :template "make chocolate chip cookies")

(defplan chop-chocolate
  :constraints ((not (has-food-processor)))
  :subgoals ((plan chop-choc-with-knife))
  :template "chop up the chocolate")
```

[Weiner 80, Mann 87] which could be constructed using planning operators, and these types of discourse might be improved using discourse markers and meta-comments.

These operators may be interpreted as describing the planning of a task, rather than how to explain it. Because the correspondence between a task and its explanation is so close, there is little distinction. However, it leaves unclear whether the decisions whether to describe sub-tasks in detail belong in the content plans or presentation plans¹⁸. If these decisions were put in the task descriptions above (consistent with the electronics content plans), it would result in an extra level of unnecessary detail:

```
(defplan chop-chocolate
  :constraints ((understands chop-chocolate))
  :subgoals ((say "chop up the chocolate")))

(defplan chop-chocolate
  :constraints ((not (understands chop-chocolate))
                (not (has-food-processor)))
  :subgoals ((plan chop-choc-with-knife)))
```

Because the decisions about what to explain in task descriptions are so uniform, they are better considered aspects of the presentation. In the example system, the following modification to the 'inform.exchange' plan deals with this.

```
(definteraction inform.exchange (plan)
  :constraints ((understands plan))
  :subgoals ((inform.move (plan))
              (acknowledge.move (plan))))

(definteraction inform.exchange (plan)
  :constraints ((not (understands plan)))
  :subgoals (plan))
```

The second added operator will call the content planner to expand the plan if it is unfamiliar to the hearer.

Some other minor changes are made to the dialogue control aspects of the discourse planner, and will be described in the next chapter. However, the majority of the system is unchanged, or just simplified. Meta-comments and

¹⁸Another approach would be to keep the domain (task) description separate from the content planning level. However, the resultant content plans would be very simple.

clue words are still included, so a typical description (given the same ‘tutorial discourse’ operators) might start with:

Now, I’m going to tell you how to make chocolate chip cookies.
You will need:
8 oz chocolate
5 oz flour
.....

4.2.4.2 Changing the Discourse Genre

In a different situation, the introduction to the cookie explanation given above would not be appropriate. Both the language and the discourse structures used are based on tutorial discourse where there are unequal social roles, and where the ‘teacher’ has control of the discourse. This section will therefore conclude the discussion of different types of discourse with a brief discussion of how the presentation of the information may be varied.

From the discussion so far, two main ways of influencing the resulting text should be apparent. First, a different set of linguistic act templates may be used for the discourse. These would have the same function, but different levels of politeness or familiarity. For example:

- (1) Now, I’m going to tell you how to make chocolate chip cookies.
- (2) OK, this is how to make chocolate chip cookies.
- (3) Right then, you’re going to make chocolate chip cookies.

The above sentences are all based on the introductory clue word and meta-comment, but reflect different roles between speaker and hearer.

Second, different discourse plans may be used. These might be used to:

- Generate a piece of text with no interactions allowed.
- Generate a straightforward explanation with no meta-comments.
- Generate an explanation allowing interruptions, but not asking the user questions.

- Generate explanations where responses to user questions and replies are handled in different ways. (cf. [Woolf & McDonald 84]).

A fuller description of all the ways the discourse may be varied is given in section 5.2.5. However, this section should make clear the advantage of separating the discourse and content plans — it allows the same textual content to be presented in a number of different ways.

4.2.5 Summary: Planning Communicative Text

The above discussion has shown how the content and presentation plans are used jointly to plan an explanation, and how these may be varied depending on the domain or type of discourse. As the explanation is planned, control may switch between planning the content and planning the presentation, but a single planner is used throughout. As planning proceeds a global data structure is built up of the evolving plan, and this is used as a simple discourse model. Content plans may take domain objects as arguments, while discourse or presentation plans take content plans as arguments, and may be considered a simple form of meta-planning. This distinction between two levels of planning is important, and one which most work on text planning fails to make — the ‘how to say it’ component typically considers only the surface structure of sentences.

The final section will show how these features contribute to satisfying the criteria introduced in section 4.1.

4.3 Conclusion

So, in section 4.2 the basic EDGE text planner was presented. Important features of the approach include the use of two levels of planning (content and discourse); the use of prerequisite and sub-skill relationships in content planning (rather than

schemas or coherence relations); and the use of an explicit representation of the evolving plan or discourse model.

The use of two levels of planning makes it easy to generate discourse with different styles, but with the same content (or vice-versa). The use of prerequisite and sub-skill relations allow explanations to be tailored to the user's prior knowledge and makes it easier to generalise the plans to other domains. The use of an explicit discourse model corresponding to the intentional structure of the discourse allows plan selection to be influenced by the immediate discourse context, and will prove vital for dealing with interruptions (described in the next chapter). It also provides the potential for improved models of pronoun selection.

The following points summarise how the original criteria in section 4.1.3 are satisfied.

Cohesion: The simple model of focus in plan arguments allows pronouns to be used both for domain objects and for explanation plans themselves.

Coherence: Ordering of subgoals and the constraints imposed by the fact that arguments of subplans must be derivable from the arguments of the calling plans imposes some coherence, though these could be made more explicit.

Intentionality: The text clearly derives from the intention of the speaker, with the intentional structure explicit.

Informativity: Alternative plans and prerequisites allow appropriate content to be selected for particular users, while the use of meta-comments and discourse markers improves intelligibility.

Situationality: It is possible to vary the presentation of the content in various ways depending on the general situation, while the discourse model can be used to select plans depending on the immediate situation.

Text Types: Two levels of planning are used to capture two types of textual organisation. Both these include some standard orderings abstracted from the analysis of text and discourse.

Interaction and Graphical Actions: Interactive discourse is possible, and it is possible to coordinate text and graphics. These will be discussed in chapters 5 and 7.

Overall, the EDGE system seems to give a better treatment of *informativity* than previous systems and shows the advantages of using two levels of planning. However it does not emphasise how *coherence* is maintained (this is almost a side-effect of the representation), and does not consider the hearer's goals at all. Further improvements in flexible text generation systems will result from making more of these principles explicit.

The planning formalism used is very simple, based on incremental expansion of skeletal plans. However, it appears adequate at least in the domains examined. Interesting extensions would include allowing partially ordered plans, allowing discourse plans to transform the content plans in various ways, and to use a more flexible, adaptive planning algorithm. The first two of these are only possible because of the explicit representation of the evolving plan. All these extensions will be discussed in chapter 8.

The next chapter will show how the EDGE system may be extended to allow interactive discourse.

Chapter 5

Controlling Interactions

In the last chapter the content planning part of the EDGE discourse generator was described. This was shown to satisfy a number of criteria for generating communicative text, taken from work on text structure. This chapter will complete the description of the discourse planner, showing how interactions may be managed within a complex explanation.

The chapter will start by examining some of the work on the structure of human dialogues, reviewed in chapter 2. These will provide the basis for a set of criteria or principles which should govern the development of the model. Next, the EDGE discourse planner will be described in detail, and finally the chapter will conclude with a discussion of how the model meets the criteria presented, and how far it contributes to models of discourse.

5.1 Criteria for Structuring Discourse

The EDGE dialogue model will be based on insights from a variety of fields — in particular, linguistic and sociolinguistic models of discourse and work on tutorial dialogues. This section will present some principles for the model based on these different areas of research.

5.1.1 Principles from Linguistic and Sociolinguistic Studies

Some general principles governing dialogue structure can be taken from the linguistic and sociolinguistic studies described in section 2.2.2. These include analyses of informal conversations (conversation analysis), more formal analyses of particular types of discourse (discourse analysis) and plan-based models of discourse used in artificial intelligence. Each of these is based on the analyses of numerous examples of human dialogues, though the methodology differs in each case.

This section aims to summarise the general principles which can be usefully abstracted from this work in developing a computational model. This will involve summarising the basic approach, discussing how far it has been applied in existing computational systems, and discussing how far it can usefully be applied to a model of explanatory discourse.

5.1.1.1 Discourse Analysis

Work on discourse analysis has shown how several types of discourse appear to have a regular structure which can be described by grammar-like rules applying at a number of levels. For example, the rules may be used to describe how different types of *exchanges* are structured, or how a particular type of *transaction* may consist of an organised sequence of exchanges. The particular rules will depend on the type of discourse, but different types share a basic hierarchical structure.

Hierarchical models of discourse structure have been used in two recent dialogue systems [Ferrari & Reilly 86, Reilly *et al* 88, Wachtel 86]. Wachtel, for example, uses a hierarchical discourse structure based on Burton's, and shows how the structure influences pragmatic constraints on the discourse. In both these cases the model is used more to aid interpretation than to control the dialogue.

For relatively formal types of discourse, where one participant has a dominant role, hierarchical models of discourse also provide an appropriate (and computationally tractable) way of structuring and organising interactions. For explanatory discourse, the particular structures observed in classroom discourse [Sinclair & Coulthard 75] may be adapted. However, a strict hierarchical model will be inadequate when the roles of the participants are more equal and the dialogue approaches normal conversation. It will only be viewed as a starting point for modelling explanatory discourse.

5.1.1.2 Conversation Analysis

While discourse analysis is concerned with describing the structure of different, relatively formal types of discourse, conversation analysis is concerned with analysing ordinary conversation. It suggests many general principles for providing local opportunities for interaction:

Turn Taking: There should be places in a sequence of utterances where control may pass to the other participant. This may be by the current speaker explicitly selecting the other participant, or by the other participant *self-selecting* and becoming the new speaker.

Adjacency Pairs: Adjacency pairs (such as ‘question-answer’) may be viewed as a type of expected exchange structure. So, following a question from the system, an answer should be expected (and possibly an answer menu displayed). However, it should not be demanded — there should be possibilities for avoiding answering.

Insertion Sequences: One case where an item in an adjacency pair is not immediately followed by its corresponding item is where there is an *insertion sequence* between them. For example, a question might be followed by another question and answer before being answered.

Side Sequences: A more general case of an ‘interruption’ is where there is a *side-sequence* such as a clarification sub-dialogue which is later resumed from. These are important in explanatory dialogues.

Opening and Closing Sequences: Particular sequences of conversational move are commonly used to negotiate the beginnings and ends of conversations. *Pre-sequences* are used to give either party the option of continuing some topic.

There has been some recent interest in applying conversation analysis ideas to HCI. For example, Frohlich and Luff describe how the approach has been used in an advice-giving system [Frohlich & Luff 89]. The emphasis is on giving the user the same sorts of possibilities for interaction that there are in human conversation (such as the types of interaction described above). In their system, conversation is locally managed with no general plan — it is the user who is dominating the interaction. However, although it perhaps goes against the spirit of conversation analysis work, the possibilities for local interaction can be incorporated into an extended hierarchical model of discourse structure. This will be the approach taken in the EDGE system.

5.1.1.3 AI Models of Dialogue Structure

AI models of discourse have generally been based on the plans and intentions of the speaker, and how these may be inferred by the hearer. Power’s system, for example, showed how a simple dialogue could be based entirely on the plans of the participants and on ways of using exchanges of different sorts (*dialogue games*) to convey information [Power 74]. Systems such as this go beyond the simple descriptive accounts described above. They begin to explain *why* certain sequences of exchanges are observed in particular types of discourse, and *how* they may be planned given a principle goal.

A plan-based approach clearly applies to explanatory text generation, and has been described in chapter 4. Here we are concerned with how far the approach is useful in structuring interactions. Is it possible, for example, to use Power's approach, and structure the whole dialogue from domain plans and exchange (dialogue game) descriptions? Boundary exchanges could perhaps be planned given a precondition that the hearer's attention must be focussed on a topic if that topic is to be understood.

Such an approach might be useful, but a simpler approach based (again) on skeletal plans seems adequate — dialogues are planned, but not always planned from first principles. These skeletal plans may be used both to describe the hierarchical structure of the discourse, and how to react to various sorts of interruptions. The model may be based on Grosz and Sidner's model of discourse structure and on Litman's use of domain and discourse plans.

5.1.1.4 Conclusion

The different approaches described above have a number of features in common. For example, all accept that, at a descriptive level, dialogues have some hierarchical structure — there is however dispute about whether this is the direct result of planned behaviour. All accept that there are common sequences of utterances, such as question-answer sequences, though these are normative rather than prescriptive. Although the three approaches are very different, they can be combined (to some extent). AI work on dialogue structure suggests a plan-based approach. The actual (skeletal) plans may be adapted from work on discourse analysis¹, while conversation analysis suggests what opportunities for local interaction should be incorporated into the framework.

¹Note that the specialisation and decomposition of skeletal plans appears similar to the notions of system and structure in systemic linguistics [Berry 75] on which much of the work on discourse analysis is based.

5.1.2 Principles from Work on Tutorial Dialogues

Work on tutorial dialogues has concentrated on formalising the strategies used by human teachers to respond to errors and misconceptions. This work both suggests strategies which may be used in a tutorial explanation system to respond to errors and misconceptions discovered, and suggests possible representations for implementing these strategies. Woolf and Murray's work, for example, includes a remediation strategy to 'acknowledge grain of truth in answer, and to teach by consequence, example or guidance' [Woolf & Murray 87]. The general approach, then, is to supply further information or questions which will lead the student to understand her mistake. Their representation is based on using a modified ATN formalism (the TACTN framework [McDonald 86]) to define schemas capturing these different strategies.

So, response strategies should include variants on the remediation strategy described above. These can be represented in the EDGE framework, which should be compared with the TACTN system.

There are two main difference between the TACTN framework and the EDGE skeletal plan framework. The first is that in the TACTN framework the conditions are on options within a schema, while in the EDGE system conditions are only allowed on the schemas themselves, to select an appropriate one. The second is that the TACTN system uses register assignment to record salient features of the interaction, while the EDGE system uses arguments to skeletal plans, the only global representations being (in general) the discourse model and the model of the user's knowledge. However, a set of TACTN schemas may always be rewritten as EDGE schemas and vice-versa, though they will result in different discourse structures². It is not clear which structures are the more 'cor-

²To rewrite a TACTN network as a set of EDGE planning operators, registers in the TACTN become either arguments in the plans or global properties of the discourse

rect' or helpful, though the EDGE system is more declarative, with less hidden in discourse register manipulation.

Work on tutorial dialogues emphasises the need for opportunistic tutoring and complex responses in a tutorial explanation system. It suggests the type of strategy which should be incorporated, and possible frameworks for implementing these strategies. The similarity between the existing framework and the TACTN system suggests that the existing EDGE framework will be suitable for implementing these strategies.

5.1.3 Summary

This section has considered how far work on the structure of human dialogues (reviewed in chapter 2) can be usefully applied to structuring interactions in explanatory discourse. From these different fields we can obtain a number of principles:

Discourse Analysis: Discourse (dialogue) structure should be described at a number of levels in a hierarchical model. Particular structures for tutorial interactions may be taken from [Sinclair & Coulthard 75].

Conversation Analysis: Users should be given the opportunity to interrupt and influence the discourse in different ways:

- It should be possible for users to *self-select* and interrupt the explanation at certain points.

model. Then the TACTN net may be traversed, with a new EDGE planning operator created whenever there is a condition on an arc. The subgoals of the initial planning operator being constructed will be the nodes traversed before this point, and the new planning operator.

- *Insertion Sequences* and *Side-Sequences* should be allowed, with the system able to resume the interrupted discourse appropriately.
- Conversation and topic openings and closings should be negotiated where appropriate.

AI Models: Both the hierarchical models from discourse analysis and the more local interactions studied in conversation analysis may be combined into a plan-based framework. The overall system should be consistent with plan-based linguistic models such as [Grosz & Sidner 85].

Tutorial Dialogues: It should be possible to define tutorial responses to errors and misconceptions.

Overall then, the model should combine global plan-based or hierarchical approaches with rules for reacting locally to specific situations as they occur. Although hierarchical discourse structures will be defined, the future discourse will not be planned in detail until required, so may take into account the changing situation. Also, various possibilities for local interaction will be provided, such as allowing interruptions such as clarification questions at many points in the discourse.

The model should also combine general methods for dialogue management, such as methods for dealing with interruptions, with specific tutorial strategies for responding to errors and misconceptions in a particular domain. All of these will be defined within the same basic framework.

The details of the system developed are given below.

5.2 The EDGE Discourse Planner

The EDGE discourse planner has two (overlapping) functions — to control interactions and to decide on aspects of the presentation. The latter aspect was introduced in the last chapter. This chapter will therefore complete the description of the planner by showing how it may be used to control interactions.

The model is based on a modified version of Sinclair and Coulthard's model of classroom discourse, simplified so that it is appropriate for one-to-one interaction and extended to allow the more flexible types of interaction observed in chapter 3. These include dealing with interruptions from the student and providing complex responses to errors and misconceptions. Of course, the general approach applies to other types of discourse, so it should not be viewed simply as a system for controlling tutorial dialogues.

First the basic model will be described (introduced in the last chapter), then the main extensions. The full model, with all the interaction definitions and a comparison with Sinclair and Coulthard's model, is given in Appendix C.

5.2.1 The Basic Model: Structuring Sequences of Exchanges

The basic EDGE model is used to structure sequences of exchanges with the user. This involves planning opening and closing exchanges, and selecting and structuring teaching exchanges on the chosen topic. The model is based on skeletal plan definitions for *transactions*, *exchanges*, *moves* and *acts*. These define the basic hierarchical structure of the discourse. The following sections will discuss how the different types of exchanges are planned.

```

(definteraction informing.transaction (teaching-goal)
  :subgoals ((interaction boundary.exchange
                    (teaching-goal 'open))
              (interaction teaching.exchanges
                    (teaching-goal))
              (interaction boundary.exchange
                    (teaching-goal 'close))))

(definteraction teaching.exchanges (teaching-goal)
  :subgoals (teaching-goal))

(definteraction boundary.exchange (teaching-goal type)
  :constraints ((eq type 'open))
  :subgoals ((interaction framing.move (type))
              (interaction focussing.move
                    (teaching-goal type))))

(definteraction boundary.exchange (teaching-goal type)
  :constraints ((eq type 'close))
  :subgoals ((interaction request-close.move
                    (teaching-goal))
              (interaction answer-close.move
                    (teaching-goal))))

```

Figure 5-1: Top Level Interaction Definitions

5.2.1.1 Planning Opening and Closing Exchanges

The top level plans were introduced in section 4.2.3.1. which described how an *informing transaction* could be defined involving opening and closing boundary exchanges, and an opening boundary exchange defined as a frame and focussing move. However, it was observed in chapter 3 that closing exchanges in the explanatory dialogues analysed have a different form. They involve negotiating the closing of a topic. The top level structuring of the dialogue therefore involves the definitions in figure 5-1.

The closing boundary exchange (the last definition in figure 5-1) involves interaction with the user. In particular, there is a goal which requires an answer from the user. Whenever the plans involve moves by the user, these are specified in the plan. The system will then put up a menu, or read in an answer at this point. In the current system there are options within the menus for delaying answering the question, allowing *insertion sequences*.

The following are typical opening and closing exchanges³:

Opening Exchange:

Framing move: Right,

Focussing move: I'm going to explain how the light detector circuit works.

Closing Exchange:

Request-close move: Is that enough about how the light detector works?

Answer-close move: **Yes.**

The other major constituents of the basic theory are the teaching exchanges. The *teaching exchanges* discourse plan transfers control to the content planning by putting the teaching goal on the goal stack. The domain-content plans in turn may have single teaching exchanges as their terminal elements. These will be described in the next section.

5.2.1.2 Teaching Exchanges

There are several types of teaching exchange which should be discussed. The first are exchanges used by the teacher to convey some information. This may be either indirectly — by asking a question and correcting the user if she gets it wrong — or directly, by telling the user and getting an acknowledgement that it was understood. These *teacher-initiated* teaching exchanges will be discussed first, and then the remaining types of exchanges.

³In all these examples the user's responses are in bold type face.

```
(defplan what-it-looks-like (device)
  :subgoals ((call pointat (device))
             (interaction teaching.exchange
              ((list 'describe-comp device))))
  :template ("what" (deviceref device) "looks like."))
```

Figure 5–2: Content Plan Calling Teaching Exchange

Teacher-initiated exchanges are initiated by the content planner. The content plan may have a *teaching exchange* as a subgoal (as in figure 5–2). This will result in control passing from planning content to selecting and planning the exchange.

The goal of this type of teaching exchange is for the system to have increased confidence that the user understands some proposition. This may be best satisfied by a question or a statement, depending on the system's current assessment of the user's knowledge. If the system believes that the user does not know the proposition, then it is best satisfied by telling the user the fact — then the system will have a low level of belief that the user knows the proposition. If the system has some level of belief that the proposition is known, then it should ask a question to check understanding. If the system believes that the user definitely understands the concept it should still tell the user if it is part of the plan — it may be part of an argument being developed, be important to maintain coherence or be important for maintaining the user's confidence.

So, a teaching exchange may be defined as follows⁴:

⁴Note that the second plan has no constraints, but because of the order in which the system searches for a plan instance it will only be called if the fact is *not* 'maybe-understood' — i.e., if it is not understood or if it is definitely known. It would arguably be better style to include all the constraints explicitly.

```

(definteraction teaching.exchange (fact)
  :constraints ((maybe-understood fact))
  :subgoals ((interaction teacher-elicited.exchange (fact))))

(definteraction teaching.exchange (fact)
  :subgoals ((interaction teacher-inform.exchange (fact))))

```

The subgoals here refer to sub-types of teacher-initiated teaching exchange, rather than to lower ranks such as moves or acts. The rest of this section will describe these two teaching exchanges in more detail, before moving on to describe the other types of teaching exchange allowed.

Teacher-Elicit Exchanges

A teacher-elicited exchange is defined in terms of its constituent moves⁵:

```

(definteraction teacher-elicited.exchange (fact)
  :subgoals ((teacher-elicited.move (fact))
             (pupil-reply.move (fact))
             (teacher-response.move (fact (answer)))))

```

This representation could have been generalised in two ways. It could have been described in terms of the general 'Initiation-Response-FollowUp' sequence of the Birmingham model, with arguments to determine the type of initiation or response. Or it could have been described in terms of a general *elicited* exchange, with arguments determining the participants. The second approach is inappropriate where the participant roles are so different, though would be appropriate for more balanced dialogues. The first approach is arguably clearer, but makes no difference in practice, and in the simple system described would necessitate extra rules for specialising these types.

⁵Note that the argument '(answer)' causes a global variable set during the pupils reply to be accessed — a better solution would be to use pattern matching in the arguments of the goals.

A teacher-*elicit* move consists of an *elicitation* act. An appropriate question template is obtained for the proposition type, and the textual question presented to the user. A pupil-reply move also consists of a *reply* act and this act presents a menu of possible answers and lets the user select one⁶. A teacher-response is a little more complex, having an internal structure of several linguistic acts:

```
(definteraction teacher-response.move (fact answer)
  :subgoals ((interaction accept.act (fact answer))
             (interaction evaluate.act (fact answer))
             (interaction comment.act (fact answer))))
```

Depending on whether the answer is correct or not, this might result in statements like:

- (1) Yes, very good, it's high.
- (2) No, not quite, it's actually quite low.

However, the *comment* act may result in a more complex response, which acts as an interruption, discussed in section 5.2.2.1 below.

Informing Exchanges

Informing exchanges consist of an *informing* move and an acknowledgement from the user. In human verbal discourse this acknowledgement may be non-verbal, including the use of eye movements and facial expressions. In human-computer discourse it may be realised variously by clicking a mouse button, clicking on 'OK', or typing the word 'OK'.

⁶Of course, the form of the interaction can be simply changed so that the system reads in a user's typed answer rather than using menu based input. Menus were chosen so that the system was robust enough to be evaluated.

Such acknowledgements are important. If human-computer discourse is to be robust, there must be some chance for people to signal misunderstanding [Hayes & Reddy 83]. One suggestion that has been made is to print out explanations at a 'slow reading speed', allowing interruptions at any point [Frohlich 88]. However, this approach would be frustrating for faster readers, or those who want to skip quickly. Requiring explicit acknowledgement at certain points is the alternative, allowing interruptions at these points, yet leaves open the question of how often to pause. This question is not adequately answered in this thesis, and probably should depend on user preferences and expertise. The approach taken is to pause after each proposition conveyed, but not after meta-comments (in boundary exchanges) which have no propositional content. However, alternative approaches may be simply defined by varying the discourse plans. For example, some users might prefer to only have pauses after the presentation of a whole topic. Also, if the hearer had a more dominant role in directing the discourse, there could be pauses in boundary exchanges too so the user had a chance to redirect the discourse.

Pupil-Elicit Exchanges

Another type of exchange to be considered is the *pupil-elicit* exchange where a student asks a question, usually to clarify some point in the explanation. This type of exchange may be initiated by the student at any point during the discourse (such as when the system is pausing for acknowledgement or for an answer). It will be treated as an interruption (discussed in section 5.2.2.1).

A pupil-elicited exchange has a simple form, consisting of a question from the student and a response from the teacher. However, the teacher need not always respond by answering the question — it may be considered irrelevant, or it may be about to be answered. So the teacher-response move has a complex definition, including rules like:

```
(definteraction teacher-reply.move (question-goal)
  :constraints ((abouttodo question-goal))
  :subgoals ((interaction avoid-question.exchange
                        (question-goal 'abouttodo))))
```

```
(definteraction teacher-reply.move (goal oldgoal)
  :subgoals ((interaction interrupting.transaction
                        (goal oldgoal))))
```

The first of these rules is one of a set which look for possible reasons to avoid answering the question immediately. The second is the normal (default) rule which initiates an interrupting transaction to answer the question.

The 'abouttodo' function will examine the goal stack to see if the question goal is already there. For example, at an early stage in the discourse the goal stack might look like this:

```
((interaction teacher-inform.act
  (describe-event LDR (light-intensity low) ...))
(interaction pupil-acknowledge.move
  (describe-event LDR (light-intensity low) ...))
(interaction teaching.exchange
  (describe-event LDR (light-intensity high) ...)))
(teach teach-causal-behaviour (light-detector))
(teach what-it-does (light-detector)))
```

If the user asks at this point about what the light dependent resistor (LDR) does when the light intensity is high, the system should recognise that it was about to say that anyway and initiate an *avoid-question* exchange. This involves telling the user why the question might be deemed inappropriate and asking if it should still be answered, for example:

I'll be getting on to that in just a moment. Do you still want it answered right now?
— No.

The teacher-reply move has an exchange as a subgoal. This is one of several cases where a simple hierarchical discourse model is inadequate to represent the

complexities of dialogue control. A reply may consist of a single informing act, or a whole transaction on a new topic, and that transaction may be negotiated with the user. However, the categories of transaction, exchange, move and act still appear useful in providing a clear, structured model of the dialogue.

Asking What the Student Knows

The final type of exchange dealt with is where the system may need to ask the user directly if she knows some fact or area. This happens if the system has no direct or indirect information in the user model about whether some concept is known, and needs this to make a decision about how to explain⁷. The form of this type of exchange is simple. The system asks whether the user knows the concept, and the user replies with either **yes**, **no** or **don't know**.

5.2.2 Extensions: Interruptions and Remediation

The previous section showed how exchanges with the user may be selected and structured in an explanatory dialogue. These were largely based on Sinclair and Coulthard's hierarchical model of classroom discourse, but could be easily adapted for other types of discourse. This section will consider two extensions to the basic model. The first examines how the discourse model which is being built up can be used in handling interruptions. This is a general extension, applying to all discourse types. The second concerns how to plan tutorial interventions or remediation sequences when misconceptions are discovered. This extension is specific to tutorial discourse.

⁷Note that this will only be appropriate in certain circumstances — for example, it would probably not be appropriate to ask whether the user knows about an analogous device when deciding how to describe some device. These cases may be flagged so that when the user model is checked the user will not be asked a direct question.

5.2.2.1 Interruptions and the Discourse Model

Several types of interaction are treated as interruptions. For example, if the system provides some remediation following some evidence of misunderstanding on the part of the user, then this is an interruption to the main flow of the explanation. In general the main explanation should be resumed once the remediation sub-dialogue is complete. Similarly, if the user asks a question then this (and the reply) is regarded as an interruption. In this case, when a question is asked, a *pupil-elicited* exchange is put on the goal stack (as described in section 5.2.1.2), which in turn initiates an *interrupting* transaction with the response plan as its main argument. In the former case the *teacher-comment* act may call an interrupting transaction with the appropriate remediation plan as its main argument. This will be discussed further in the next section.

Interruptions require some introduction and some way of resuming the main discourse. For simple interruptions, 'push' and 'pop' discourse markers may be adequate, resulting in interrupting sequences like:

Well, the light intensity is high so the resistance is low. Anyway

However, for longer interruptions more work is required. One way of resuming from a long interruption is with a meta-comment like:

Anyway, we were in the middle of explaining how the light detector works.

Another way commonly used in verbal explanations is to repeat part of what was said before the interruption. The following example is taken from the transcripts of verbal expert-novice explanations:

E: Each of these acts rather like a switch, so if this is high..

N: You say shorted back to base, but there's a capacitor?

E: Right, there is indeed, come to that in a sec.

Each of these is acting like a switch, so if this voltage here is high, this will turn on.

In some cases a combination of these strategies could be used, possibly involving an exchange with the novice, for example:

Anyway, I was just in the middle of explaining how to chop up the chocolate. Shall I go through that again?

For most of these examples the system must be able to work out what it was in the middle of doing — then the appropriate meta-comment may be accessed or utterances repeated. For this, the discourse model must be examined. This is a complete representation of the explanation plan where each goal includes pointers to its subgoals and parent goal. It represents the intentional structure of the discourse (with the arguments to plans giving a primitive model of focus) [Grosz & Sidner 85]. Discourse models of this sort have been discussed by Schuster and others [Schuster 88]. It is also similar to the global curriculum plan used by Murray and others [Murray 89].

A simplified portion of a typical discourse model is given in figure 5-3. This might be the state of the discourse model after the initial opening statement, but before it began explaining what the light detector does when the light intensity was high. The starred goal indicates the current position in the discourse. As the discourse progresses, the subgoals after this goal (which are on the goal stack) are expanded, but at this point there is only an outline plan of the future discourse.

In this figure the indentation is used to show the *dominance* relationships between goals, and therefore the intentional structure. The attentional state is represented by the arguments of the ancestors of the current goal, so at the starred goal would include:

```
(teach how-it-works (light-detector))  
  (light-detector)  
    (light-detector (light-intensity high))  
      (suppose (light-detector (light-intensity high)))
```

As in Grosz and Sidner's representation of attentional state, this includes the current proposition being expressed (suppose), the objects being talked

```

(interaction informing.transaction
  ((teach how-it-works (light-detector))))
(interaction boundary.exchange ((teach ....) 'open))
  (interaction frame.move ('open))
    (interaction marker.act ('open))
      (interaction focussing.move ((teach ....)))
        (interaction meta-comment.act ((teach....)))
          (interaction teaching.exchanges ((teach ....)))
            (teach how-it-works (light-detector))
              (teach causal-behaviour (light-detector))
                (teach particular-causal-behaviour
                  (light-detector (light-intensity high)))
                  (interaction teaching-exchange
                    ((suppose light-detector (light-intensity high))))
                    *** (interaction teacher-inform.exchange ((suppose ....))
                      (teach causal-bit
                        (light-detector (light-intensity ....)))
                        (teach particular-causal-behaviour
                          (light-detector (light-intensity low)))
                          (teach what-it-does (light-detector))
                          (interaction boundary.exchange ((teach ....) 'close))

```

Figure 5-3: Example Discourse Model

about (the light-detector and the input value), and the discourse segment (the plan to teach how the light-detector works)⁸. More distant ancestor goals will be less salient than recent ones.

This representation can be used to find out which goal the system is 'in the middle of' doing. The discourse model must be examined to find which ancestor goals of the next goal on the stack led to utterances in the previous discourse section (before the interruption). The lowest such goal is the one the system should say it was in the middle of executing. To find such a goal requires an explicit discourse model — a tree representation of the whole discourse, rather than a list of propositions uttered or a stack of goals.

Interruptions are structured by defining an interrupting transaction. The basic form of the interruption is defined as follows:

```
(definteraction interrupting.transaction (goal)
  :subgoals ((interaction boundary.exchange (goal 'open-int))
              (interaction teaching.exchanges (goal))
              (interaction boundary.exchange (goal 'close-int))))
```

In the current system the closing (resumption) boundary exchange normally involves a discourse 'pop' marker such as 'anyway' and a meta-comment on the interrupted discourse (e.g., "Anyway, I was in the middle of explaining what the LDR does"). However, if the interruption came after a question the resumption will involve re-asking the question. The following is an example dialogue section illustrating this sort of resumption (taken from the scripts of the evaluation session):

⁸The attentional state could be extended to include all the ancestor goals/discourse segments. However, only including the arguments to plans means that only discourse segments that are explicitly focussed on are included.

What's it going to be when the input light intensity (L1) is low (dark)?

— **What's a resistor?**

OK, this component here is a resistor.

Anyway, what's the output voltage (V1) of the light detector circuit going to be when the input light intensity (L1) is low (dark)?

— **Quite low (towards 0V)**

Although remediation sub-dialogues and interrupting questions are both structured using the 'interrupting' transaction, they are treated differently in the discourse model. The remediation sub-dialogue is not a true interruption — it shares an *intention* with a preceding discourse segment [Grosz & Sidner 85, pg 40]. So though it interrupts the main flow of the explanation, the *attentional state* is treated exactly as if it were part of the main explanation. On the other hand, user questions do not necessarily share a discourse intention⁹, and will not be added to the main discourse model. By only adding such interruptions to the goal stack, and not to the main discourse model, a stack of discourse models is built up, as suggested by [Litman 85], and items in focus will be restricted to those in the interrupting segment¹⁰.

The discourse model obviously has many potential applications apart from dealing with interruptions. It provides a better basis for selecting referring expressions (though it is not currently used for that — just a simple representation of the items in the preceding utterance). It also allows the system or user to

⁹User questions **may** share an intention with the speaker (such as for the user to understand some information) — however, this intention may not be the immediately preceding one. Rather than attempting to work out what the dominating intention is, they are currently always treated as true interruptions.

¹⁰So that knowledge about interrupting questions is not lost, the interrupting discourse segments are kept on a list of questions answered. This list is used to avoid repeating things which have been already been explained.

refer to the past discourse in terms of the discourse goals, for example, asking for a repeat (“Can you explain what the light detector does again.”) or mentioning that something has just been explained (“Well, we went through that a minute ago.”). However, this section should at least have shown how it is used for dealing with interruptions, and how it provides a representation of intentional and attentional structure consistent with Grosz and Sidner’s model.

5.2.2.2 Opportunistic Tutoring and Remediation

The final extension to the basic discourse framework is to allow complex remediation sub-dialogues when a user gets answers to questions wrong. This is a specific extension for tutorial type explanations, and allows dialogues such as those described by Woolf and Murray [Woolf & McDonald 84, Woolf & Murray 87].

Remediation strategies are called from the *teacher-comment* act which is part of the *teacher-response* move. If the user has got an answer to a question wrong, and there is a response or remediation strategy which can be used, then this act will cause an *interrupting* transaction to be called with the appropriate remediation plan as its argument. If there is no such plan, then the act will cause the correct answer to be given. The following plan definition shows how the interrupting remediation strategy is called — the response function searches the set of possible remediation strategies applying to that type of fact for one that has all its constraints satisfied.

```
(definteraction teacher-comment.act (fact answer)
  :constraints ((response fact answer))
  :subgoals ((interaction interrupting.transaction
                ((response fact answer)))))
```

So, the default response to an incorrect answer is to signal it incorrect, ‘evaluate’ the answer and then give the correct answer. This is consistent with Murray and Woolf’s remediation schema. However, as in their schema, preference is given to providing some more concrete remediation if that is possible. Murray

and Woolf include three possible remediation actions — giving examples, giving hints or discussing the consequence of the answer given. These different schemas may be activated in different contexts.

In the EDGE system, remediation plans are defined in the same way as the main discourse or content plans. For example, the following plan is used when the user answers an input-output question incorrectly, but knows the behaviour of an analogous device. A comparison with the analogous device will be made and the question asked again.

```
(defplan event-reply (device inputs output)
  :constraints ((understands (list (getslot device 'analogy))
                                'what-it-does))
  :subgoals ((teach similarity (device
                                (getslot device 'analogy)))
              (interaction marker.act ('conclusion))
              (interaction teacher-elicit.exchange
                ((list describe-event device
                       inputs output))))))
```

It would be possible to define a more domain-independent remediation plan which would be specialised for particular problem types. The above example would be an instance of the 'give a hint and ask again' plan. However, the simple definitions such as the one above illustrate sufficiently how the system handles tutorial responses.

The above rule might lead to the following remediation sub-dialogue. This example is again taken from the scripts of people using the system in the evaluation sessions.

What's the output voltage (V2) of the heat detector circuit going to be when the input heat intensity is high (hot)?

— **Low (0V).**

Wrong, not quite,

OK, A heat detector circuit is much like a light detector circuit only:

– Its output voltage depends on the input light intensity.

– It has a thermistor instead of a light dependent resistor.

This component here is a thermistor.
Let's go through what a thermistor does.
When the input heat intensity (H1) of the thermistor is high (hot) the output resistance (R1) will be quite low.
When it is low (cold) the output resistance (R1) will be quite high.
So, what's the output voltage (V2) of the heat detector circuit going to be when the input heat intensity is high (hot)?
— **Quite high (towards 9V).**
Precisely, very good, it's quite high (towards 9V).

5.2.3 Limitations of the Model

There are a number of outstanding problems that should be discussed, where the existing representation is either inadequate or at least inelegant. The main problems relate to planning ahead. First, the basic algorithm involves calling content prerequisite goals if the associated concept is not believed to be known. Yet the system might not know whether the user knows this, or it might change before that goal was actually executed. It is therefore necessary to put a conditional goal on the stack, such as:

```
(if (not (understands (light-detector) what-it-does))  
    (teach what-it-does (light-detector)))
```

The existence of conditional goals make it harder to predict whether a topic is about to be discussed. If that topic is conditional on the user's understanding then the best that can be done is to evaluate the condition in the current context and see whether, if the user's believed understanding does not change, that topic would be introduced.

Another issue is how to decide what meta-comments and discourse markers are necessary. Intuitively, this seems to depend on the length or complexity of the relevant discourse segment. But this cannot always be determined beforehand as the discourse may not yet be planned. The simple solution used is to explicitly state in content plans whether this should be a sub-transaction, as shown below, yet this mixes discourse and content knowledge.

```

(defplan component (device)
  :preconditions ((what-it-looks-like (device)))
  :subgoals ((interaction sub.transaction
                        ((plan 'teach 'what-it-does (list device)) 'sub)))
  :template ("what" (deviceref device) "is like"))

```

The current model also does not address the problem of natural language interaction. Questions and answers must be selected from a menu, or entered in a formal language. Yet the discourse model provides strong constraints on the linguistic acts expected, as well as the items which would be referred to. The framework therefore provides a good basis from which to develop a natural language understanding component which incorporates both global focus and dialogue context information, using a combination of the ideas of Grosz, Litman and Allen and Wachtel [Grosz 77, Litman & Allen 84, Wachtel 86].

5.2.4 Example Interactions

In chapter 4 a brief example was given showing how the overall discourse is planned, beginning with a goal such as:

```

(interaction informing.transaction
  ((teach how-it-works (light-detector))))

```

In this section, detailed examples of two teaching exchanges will be given, illustrating how exchange structures, remediation plans and interruptions are used. The first example illustrates how system-initiated teaching exchanges are selected and structured, including possible 'remediation' planning.

System-Initiated Exchange

Given a goal such as:

```

(interaction teaching.exchange
  (describe-event LDR (light-intensity high) (resistance lowish)))

```

the system will reason which sub-type of teaching exchange should be used. Depending on whether the proposition is believed understood then either an *informing* exchange or a *teacher-elicit* exchange may be selected, and one of the following utterances generated:

- (1) When the input light intensity of the light dependent resistor is high, the output resistance will be quite low.
- (2) What's the output resistance of the light dependent resistor going to be when the input light intensity is high?

An informing exchange expects an acknowledgement from the user. Once this is obtained then the explanation planning and generation will continue. However, suppose the teacher-elicite exchange was selected. This consists of a question, answer and response. In order to obtain an answer a menu of possible replies is presented and an option selected by the user. If the answer is correct then the system's response will be something like:

Yes, very good, it's quite low.

However, suppose the user replies with 'Low — fully conducting' (selected from a menu). This does not match with the correct answer, so the system will look for a remediation strategy. If, for example, the user knows what a thermistor does, it will generate the utterances:

No, not quite,
OK, a light dependent resistor is much like a thermistor only:
– Its output resistance depends on the input light intensity.
So, what's the output resistance of the thermistor going to be when the input light intensity is high?

However, if there is no suitable remediation strategy then the following is generated:

No, that's not quite right, it's actually quite low.

User-Initiated Exchange

The next example illustrates how the system responds to user questions. If the user interrupts at some point with a question, this initiates a pupil-elicited exchange. For example, suppose the user asks towards the beginning of the dialogue:

What does a light dependent resistor do?

The system may decide one of several things. It may note that it is about to say that later anyway, that it was in the middle of saying that, or that the question is irrelevant to the current discussion¹¹. Suppose the system was about to explain what a light dependent resistor does anyway. The system will then state the reason for avoiding the question and ask if it should still be answered:

I'll be getting on to that in just a moment. Do you still want it answered right now?

If the user still wants it answered immediately then an interrupting transaction is initiated to answer the question. This results in the following sequence of utterances:

Ok, I'll go through what a light dependent resistor does.
When the input light intensity of the light dependent resistor is high,
the output resistance is quite low.
When it is low, the output resistance is quite high.
Anyway, we were describing the structure of the light detector circuit
....

¹¹The current system does not use the last of these as it seemed likely that the user would better be able to assess the relevance of a question to her needs than the system. However, originally relevancy was determined using a notion of implicit focus similar to that described by Grosz [Grosz 77]. A statement was considered relevant if some item in the statement had a direct link (in the domain model) to some item in current global focus.

Later in the explanation the system may be about to explain something which has already been answered because of a user question. The current strategy is to miss out that bit of the explanation (pruning the plan). If the content plans were partially ordered better strategies would be possible. Then, if the user asked something which the system was about to explain, it could reason whether that goal could (according to the ordering constraints) be brought forward and answered immediately.

A more complete example discourse will be given in chapter 7, once the user modelling aspects of the system have been discussed.

5.2.5 Generating Other Discourse Types

The above examples have been concerned with tutorial explanations of simple electronic systems. However, the approach generalises to other types of discourse and different domains. In the previous chapter a version for generating task dialogues about how to make cookies was discussed. This involved modifying the content plans to represent how to make cookies, and modifying the discourse plans to include one specialised for presenting task descriptions. In general, there are five ways of changing the discourse depending on discourse genre:

- Changing the content plans and templates allows different things to be explained. The content plans define the decomposition of the subject into sub-topics. In many cases it is expected that this would be all that would be necessary to generate explanations of different subject matter but of the same type (e.g., tutorial or help), with the acts and discourse plans fixed.
- Changing the way content plans are interpreted and selected may be necessary in different situations. Moore, for example, uses a set of heuristics for selecting a plan given a set of possible plans (ways of explaining something) [Moore 89]. These include preferring operators which make no assumptions about the hearer's beliefs. In the EDGE system, only operators making

no such assumptions are currently allowed. What heuristics are used will depend (among other things) on the discourse genre. In some, a short explanation will be preferred over one that explains in great detail anything that might not be understood, and prerequisite information might not be included unless definitely unknown.¹²

- Changing the discourse plans allows different discourse structures and exchange types. The structure of different types of discourse has been examined by discourse analysts, and it would be expected that help dialogues, tutorial dialogues and consultation dialogues would all have different structures, with different types of exchanges, for example. The social roles of the speaker and hearer would influence the discourse (exchange) structure, with more balanced speaker-hearer exchange types where the social roles are similar.
- Changing the remediation plans (in a tutorial discourse) allows an exploration of different tutorial approaches. Details of the discourse plans may also be changed as the social roles of speaker and hearer are viewed differently in different tutorial approaches.
- Changing the words used in the discourse acts allow different 'attitudes' between speaker and hearer, such as polite, rude, patronising or friendly. Note that you can be polite, rude, patronising or friendly while maintaining a particular social role. There are traditional teachers representing all these types. Of course, syntactic structure and lexical choice are influenced by

¹²Of course, this also depends on the system's capability to re-explain in a different or more detailed way if the user doesn't understand. The emphasis of Moore's work is on these follow-up questions and explanations (as described in chapter 2). A system without these capabilities, and the ability to deal with interruptions, would have to be much more careful in constructing an understandable answer.

many factors, including the discourse structure and focus, the hearer's knowledge and the roles of the participants. There is no scope to consider all of these — however it is worth showing that some simple changes at the sentence and lexical level may reflect very different attitudes.

To illustrate some of these points, the 'cookie explainer' has been modified to illustrate some discourse types. First, we can see what a (slightly patronising) teacher discourse might look like:

Right then, Alison, I'm going to explain how to make chocolate chip cookies.

Do you have all the ingredients?

— Yes

Good. Well, first you should chop up the chocolate.

Do you know how to do that?

— No

Never mind, I'll explain that to you.

Do you have a food processor?

.....

If the mode of the discourse was different (e.g., non interactive), or the roles of speaker and hearer did not allow the speaker to ask questions, then a simple textual explanation might be generated:

Right, Now I'm going to explain how to make chocolate chip cookies.

First, make sure you have all the ingredients.

Now you have to chop up the chocolate.

If you have a foodprocessor, you can chop up the chocolate in it.

Otherwise, just use a knife.

A less polite discourse is the following. This uses exactly the same discourse plans as the first example, but different acts.

Ok, sc*mb*g, I'm going to tell you how to make cookies.

You have all the ingredients?

— Yes

Ok, First, chop the chocolate.

Know how to do that?

— No

Cr*t*n, I'll have to tell you then

The range of possible cookie explanations is almost unlimited. It depends chiefly on the roles and attitudes of the speaker and hearer. Roles tend to affect the discourse plans, while attitudes affect the linguistic acts used more, though this distinction is by no means clearcut. There is enormous scope for further work in this area, but I would claim that by separating the discourse plans, content plans, remediation strategies, and how the plans are interpreted provides a good basis for such explorations, as illustrated by the simple examples above.

5.3 Conclusion: Local Interactions in a Planned Explanation

Section 5.2 has shown how interactions can be planned in a complex explanation. This involves using a simple hierarchical model to organise the basic sequence of exchanges with the user, while allowing for various types of local responses. The overall hierarchical model allows clearly structured discourse to be generated and a simple discourse model to be built up. Allowing local responses (interrupting questions and remediation sequences) provides for a far more robust explanation where misunderstandings may be cleared up as they occur. The hierarchical discourse model helps in the structuring of these local interactions by providing a representation of the discourse situation¹³.

This final section will conclude by discussing how far the model meets the criteria introduced in section 5.1 and assessing its contribution as a model of explanatory discourse.

First, the principles introduced in section 5.1 will be discussed in turn:

¹³This point is developed in [Cawsey 89a] which considers how far the local interactions discussed in work on conversation analysis may be incorporated in a plan-based model.

Discourse Analysis: The model developed is consistent with the sorts of hierarchical models which have been suggested in work on discourse analysis, and in particular with that described in [Sinclair & Coulthard 75]. However, various simplifications and extensions have had to be made, such as allowing an answering move to consist of a whole sub-dialogue. This extension is consistent with suggestions by [Carlson 83] for example.

Conversation Analysis: Local interactions of various sorts are possible, in particular:

- Interruptions are possible at various points, allowing the user to ask questions. These points correspond to the *transition relevance points* discussed in work on conversation analysis.
- These interruptions (*side sequences* or *insertion sequences*) can be resumed from using the discourse model representing the discourse situation.
- Topic closings are negotiated, allowing follow-up questions.

AI Models: The discourse model developed is consistent with Grosz and Sidner's model of discourse structure [Grosz & Sidner 85]. This includes *dominance* relations between *intentions* or goals and a simple model of *attentional structure* based on the arguments to plans.

Tutorial Dialogues: The model allows tutorial responses or remediation strategies to be defined and these may be incorporated in the basic model of explanatory discourse.

The EDGE discourse model makes a number of contributions to work on discourse and text generation. First, it combines ideas from a number of fields as shown above. In particular, sociolinguistic models of dialogue structure have been integrated with Grosz and Sidner's model of discourse structure via the topic structure of the domain. The representation of attentional state allows

pronouns to be used to refer to domain objects and to discourse segments. The representation of intentional structure allows improved use of discourse markers and meta-comments, and this is consistent with Sinclair and Coulthard's model. The representation of different types of exchanges allows the dialogue to be appropriately structured.

Next, unlike many previous models (which view a dialogue as a sequence of short monologues [Sinclair & Coulthard 75, Coulthard & Montgomery 81, pg 31]), dialogue and monologue may be described within the same framework. The EDGE approach accepts that even monologues are designed interactively, taking account of possible audience reaction [Edmondson 81, Carlson 83], and that in a dialogue textual coherence applies across the whole discourse, rather than just across any single utterance. Models of dialogue (e.g., [Wachtel 86]) and monologue (e.g., text generation models) apply equally. When an explanation is given, so long as there are opportunities for interruption, it can be viewed as a sequence of exchanges, where the non-interruption at a possible 'Transition Relevance Point' is taken as a non-verbal form of acknowledgement. Other exchanges take place within the overall structure of the explanation, which is much the same structure as observed in monologue.

Finally, it provides a starting point for exploring how discourse structure depends on discourse register [Halliday 73]. Changing the *field* or topic involves using different content plans, while changing the *tenor* involves modifying the discourse plans (for example, allowing different exchanges depending on the social roles) or changing the form of the linguistic acts depending on the attitude of the speaker (polite, patronising, rude, encouraging etc). The same basic framework may thus be used to generate a wide variety of discourse types. A particular discourse genre may be partially defined in terms of the attitudes and social roles of the participants (and possibly by the subject matter addressed). Loading the

appropriate content plans, discourse plans, remediation plans and acts will result in a particular type of discourse¹⁴.

Overall then, the EDGE model makes a significant contribution to computational models of discourse, and perhaps some contribution to our understanding of discourse in general. The next chapter will complete the discourse framework, discussing how the user model may be inferred as the discourse progresses, and how this affects the discourse itself.

¹⁴To complete the discussion of discourse register it would be necessary to consider the effect of the mode of interaction. The EDGE framework is based largely on models of human verbal face-to-face interaction. Yet it is being designed for human-computer written interaction. This is a very important issue, discussed by Suchman [Suchman 87] yet inadequately addressed in current research. Even those who base their work on studies of humans communicating via a computer link fail to capture all aspects that characterise human-computer interaction.

Chapter 6

Updating the User Model

In the last two chapters it was shown how a model of the user's knowledge of the domain affects both the dialogue structure and the content of the interactions. This knowledge will be dynamically changing as the dialogue progresses. In this chapter, various approaches to user modelling will be reviewed, and the EDGE user modelling component presented. This component is integrally related to the discourse structure and interactions, and allows a dynamically changing set of assumptions about the user's knowledge to be maintained. The model is influenced by, and influences several aspects of the discourse.

6.1 Review: Inferring the User Model

One of the first systems to use some kind of user model to guide dialogue was Carbonell's 'Scholar' system [Carbonell 70]. Each concept in the domain model could be tagged 'Known' or 'Unknown', and this could influence which topics were introduced in the dialogue. This basic approach became known as *Overlay* modelling, based on the assumption that a student's knowledge could be represented as an 'overlay' of the expert's knowledge. This approach has since been used in numerous systems, despite the doubtful basis of this assumption.

In this section, the development of such models will be discussed. More complex user models, such as models of misconceptions, goals or complex beliefs will not be considered, as these are not used in the EDGE system.

A major problem in overlay modelling is how to obtain a lot of information about the student's knowledge from a few interactions. The following sections will consider some of the approaches to this problem.

6.1.1 Stereotypes

One important technique for 'guessing' the user's knowledge or preferences is to assign the user to one or more predefined stereotypes. In Rich's system for example [Rich 83] information given by the user may trigger a set of stereotypes (such as feminist, intellectual and scientist) which in turn have associated normal features (such as liking Science Fiction books). These features are used by the system to select novels the user may like, and to print out descriptions of these books which include features relevant to the user.

This type of stereotype model has the disadvantage that a great deal of work must be done encoding a set of stereotypes. In some domains a much simpler approach is possible. If users may be categorised according to their level of expertise, and difficulty levels assigned to the various concepts in the domain, then this allows assumptions to be made about what concepts will be known by the different users. This is essentially the approach taken by Wallis and Shortliffe in the Mycin explanations [Wallis & Shortliffe 84]. Each rule and concept in the knowledge base was tagged with its difficulty (or complexity) and its importance. Then, based on the expertise level of the user, rule trace explanations could be tailored to not use concepts unknown to the user or of low importance¹.

¹Note that in this system explanations are tailored to only use rules involving concepts known to the user (resulting in shorter explanations for novices), whereas in many

6.1.2 Inference Rules and Conceptual Structure

Sleeman extended this approach in his 'User Modelling Front End' (UMFE) system [Sleeman 85]. As well as using the difficulty levels of domain concepts, the approach allowed inference rules related to the conceptual structure of the domain. For example, given a hierarchy of concepts and specialisations of those concepts, inference rules could be used to deduce that, if a concept is known, then its parent and sibling concepts are probably known. When an explanation is requested, the system may ask the user a relatively small number of questions about her understanding in order to obtain enough information to provide an appropriately tailored response.

The UMFE system, therefore, may obtain information about the user's knowledge from three sources:

- From questions asked explicitly about what the user knows.
- From inferences based on the user's level of expertise and the difficulties of the concepts. The level of expertise may in turn be derived from the difficulties of concepts believed known.
- From inferences based on the conceptual structure of the domain — in particular from *general-specific* relations between concepts.

These different sources are considered in order — direct knowledge from the user first, then using difficulty levels, then conceptual structure. If there is a conflict at one level, then the system will count this as having no information about whether the concept is known.

other systems (such as [Weiner 80]), explanations are tailored by missing out justifications or facts already known (resulting in longer explanations for novices). The EDGE model, using prerequisites and constraints on plans, may miss out known (prerequisite) information, yet select the level of causal explanations to use known concepts.

There are several limitations with the system. When the user asks for an explanation, for example, the system may ask several questions about her understanding before answering the question, leading to an annoying and unnatural dialogue. Asking the user such direct questions in this context should be seen as a last resort. The system also fails to update the user model in any way based on what the user has been told, assuming that the user's knowledge will be fixed in a particular session.

Chin uses a similar approach in the KNOME user modelling system [Chin 88]. This was developed as part of the Unix² consultant project at Berkeley which aimed to develop a natural language advice system for Unix. The KNOME user modeller uses the level of expertise of the user (deduced from the concepts they appear to know) and the levels of difficulty of the concepts in the domain to provide indirect inferences about the user's knowledge. What it does not do is use the structure of the domain. Neither specialisation, prerequisite nor sub-skill relations may be used in making inferences. Concepts may only be represented at one level of coarseness so it is impossible to say that a command is known in general, but that one of the options is unknown, for example. However, the KNOME system allows a wide range of direct inferences from the different dialogue moves possible, such as assuming that the user knows any presuppositions in a question, but does not know the answer to a question.

For each concept in the KNOME system, it may be *false*, *unlikely*, *unknown*, *likely* or *true* that the user knows that concept. There are four classes of user (levels of expertise) and three knowledge difficulty levels. In this respect (and in the way it uses direct and indirect inferences) the system is very similar to the EDGE user modeller, though developed independently. The advantage of the EDGE modeller is that it can use the conceptual structure of the domain and represent concepts at various levels of abstraction.

²Unix is an operating system. "UNIX" is a trademark of AT&T Bell Laboratories.

6.1.3 Observing the User

The above discussion has considered how knowledge about the user can be obtained from initial stereotypes, from interactions with the user and from inferences based on the conceptual structure of the domain and the difficulties of the concepts. However, in some types of application more indirect approaches are necessary, where the system can only 'look over the shoulder' of the user in order to infer what they know.

Several early ITS systems used the user's actions as data for the user modelling component and could deduce what rules the user was using based on this information. For example, Clancey (in the Guidon project) uses the user's questions in a medical consultation as evidence for which rules are being considered by the user in making a diagnosis [Clancey 83a]. Burton and Brown similarly use the circuit measurements made by the user [Brown *et al* 82], while in the WEST system, the user's moves in a simple game may be compared with the experts in order to deduce which rules and strategies are known [Burton & Brown 82].

Kass and Finin present a more exhaustive set of indirect inference rules which may be used in a natural language dialogue [Kass & Finin 87], entirely using inferences based on the user's utterances. These include rules based on principles of communication, the structure of the domain knowledge and the actions performed by the user. The domain knowledge rules are extensions of the sorts of rules used in the UMFE system, while the action rules are similar to the rules described above, such as for the Guidon system. The communicative rules are inspired by Grice's maxims for communication [Grice 75], and include assuming that an utterance by the user will be considered (by the user) to be relevant to her goals and containing sufficient and non-ambiguous information.

6.1.4 Degrees of Belief

The system's knowledge about the user's understanding may never be certain. Some sources of information are more reliable than others. Many user modelling systems therefore use some 'degree of belief' that the user understands each concept (as in the KNOME system described above). This may be revised as new information is obtained. There is confusion in some systems whether this refers to the degree to which users understand a concept, or to the degree the system believes the user understands it. In all later discussion it will always be used to refer to the system's belief — it will be assumed that the user either 'understands' a concept or doesn't.

6.1.5 Summary: Sources of Information

Kobsa and Wahlster summarise the sources of information for user modelling in a dialogue system as the following [Wahlster & Kobsa 86]:

- Default reasoning from stereotypes.
- Initial models from previous sessions.
- Explicit (direct) inference from the user's input.
- Implicit (indirect) inference from the user's input.
- What the system tells the user.

Each of these categories may include a large number of types of inference. In particular, direct inference from the user's input should include inferences based on questions, answers to questions, and statements. Implicit inferences include the domain knowledge, action and communication rules described by Kass. A user modelling component should consider all these potential sources of information.

6.2 Criteria for a User Modelling Component

In general, a user modelling component should be able to make inferences about the user's knowledge based on the ongoing dialogue (as discussed above), deal with contradictions and continually changing knowledge, and provide those distinctions in the model which are required by the system which uses the user model. These three criteria will be considered briefly below.

6.2.1 Indirect Inference

A user modelling component which kept on asking the user what she understood would be unnatural and clumsy. However, a system which never asked such questions might be doing more work than it needed, and be liable to error. In general then, the system should make as many inferences as possible from the normal interactions with the user, but fall back on asking the user directly when it has no other information, or if the information is contradictory. If the dialogue is not to seem incoherent, then questions should be seen by the user to be relevant to the current discourse context. They should not be asked ahead of time while the discourse is being planned.

The system should be able to use all five sources of information discussed in section 6.1.5. Any interaction with the user should be seen as a potential source of information and not just questions asked by a user modelling component! The following direct and indirect inferences should be used:

- Direct inferences based on what the user says and asks, or on the questions she answers.
- Direct inferences based on what the user says she knows.
- Direct inferences based on what the system tells the user.

- Indirect inferences based on the user's level of expertise (possibly derived from the discourse) and on the relative difficulties of the domain concepts.
- Indirect inferences based on relationships between concepts in the domain.

Other types of inference, such as the 'communication' rules of Kass, would become appropriate if the user was allowed to communicate in natural language.

6.2.2 Dealing with Contradictions and Changing Assumptions

None of these types of inference provides certain information about what the user knows. However, information gained directly from the interactions (direct inferences) is generally more certain than information gained using complex inference rules (indirect inferences using existing assumptions). A user model representation needs some way of distinguishing between fairly certain and much less certain justifications, and ordering the possible inference rules so that the more certain are used first.

When the user model is dynamically changing, revising one small part of the model may invalidate large sections of it, as inference rules may have used the old data to indirectly derive new assumptions. If the inference rules are applied whenever new data is added to the user model (forward chaining) then this means including the justifications for features in the user model, as well as the degree of belief. One approach would be to use some kind of truth maintenance system, such as de Kleer's assumption based truth maintenance system [deKleer 86]. However, a simpler approach is to use the inference rules by backward chaining when the system needs to know if the user understands some concept. The GUMS system [Finin & Drager 86] takes this approach, using a default logic to represent facts and inference rules.

6.2.3 Using and Inferring User Model in Discourse

The final principle for a user modelling component is for it to distinguish in the model those things which the discourse system can use to influence its behaviour. The majority of work on user modelling either assumes a user model of some sort, and considers how this will influence the resulting discourse, or infers a user model which is never used to influence the discourse. Yet the goal of inferring a user model may affect the discourse structure, and the discourse will influence what can be inferred in the user model. It is thus useful to consider the use and inference of a user model together, so that any knowledge about the user which is needed to tailor the discourse may be inferred by the user modelling component as the discourse progresses.

This principle influences the user model representation selected. If the discourse system requires knowledge about the user at different levels of detail or abstraction, the user modelling system should support this. If the discourse system can use different degrees of certainty about the user's knowledge, these should be represented in the user model.

6.2.4 Summary

In summary then, a user modelling component to a dialogue system should have the following features:

Direct Inference Rules: The system should be able to make inferences about the user's knowledge both from what the user says and asks and from what the system tells the user. This should include inferences based on:

- the questions the user asks.
- the questions the user answers.
- what the user says she knows (possibly following a direct question such as 'Do you know about transistors?').

- what the system tells the user.

Indirect Inference Rules: The system should also be able to use more indirect sources of information to make guesses about what the user knows. These should use:

- relationships between concepts in the domain.
- difficulty levels of concepts and the user's level of expertise.

Dealing with Changing Assumptions: Indirect inference rules should be applied only when needed, and that information not stored. This (partly) avoids problems with changing assumptions about the user.

Using and Inferring Model: The distinctions made in inferring the user model should be those needed in the dialogue system. The discourse may be influenced by the goal of updating the user model, as well as by the inferred model. The use and inference of a user model can therefore not be separated.

Representation: It should be possible to represent concepts at different levels of abstraction, and for the system to 'believe' to varying degrees that the user understands these concepts.

6.3 The EDGE User Modelling System

In this section a simple system will be described which meets all these criteria. It is based on an extended overlay model which allows different types of indirect inferences. First, the representation and inference rules will be described. This will be followed by a summary of how the user model affects the discourse, and an example showing how the user model is updated and used in a short explanation. Finally, other types of discourse will be discussed to see how general the approach is.

6.3.1 Inferring the User Model

First, this section will show how the user model may be updated as the dialogue progresses. This will involve discussing the basic representation of the user model and showing how this is updated. The representation is based on a hierarchy of concepts based on the plans used to explain them. The system may have various degrees of belief that the user knows these concepts. Inference rules are used to update the user model based both on direct evidence (such as the user answering a question correctly) and indirect evidence. The most important type of indirect evidence used is the level of expertise of the user and the level of difficulties of the concepts. These different aspects of the system will be discussed in turn.

6.3.1.1 Representing Levels of Abstraction in Domain Knowledge

In explanatory discourse, the intentional structure of the discourse is closely related to the conceptual structure of the domain³. The user model can therefore be based on the domain plans, and use the prerequisite and subgoal (sub-skill/concept) relations in the plans to make indirect inferences about the user's knowledge. The domain plans define a hierarchy of concepts, from the most general (e.g., 'How a light detector circuit works' or 'How to make cookies') down to the most specific (e.g., 'What the resistance of an LDR is when the light intensity is high' or 'How to chop chocolate in a food processor'). Such a hierarchy of concepts is important in both inferring details of a user model (cf. [Sleeman 85]) and in using that model in tailoring the discourse. It's no good just knowing whether the user knows detailed concepts when making high level decisions about explanation strategies, yet these details are important for lower level decisions.

³Here 'conceptual structure' refers to the sub-skill and prerequisite relations rather than specialisation hierarchies (see 6.3.1.6).

The EDGE system therefore extends the overlay paradigm to allow descriptions of domain knowledge at different levels of abstraction. These levels are defined by the same plans which are used in determining the content of the discourse. Where a content plan takes arguments, the user's knowledge of that concept may be defined for each of its possible arguments, as in the following example:

particular-causal-behaviour understood:

```
((light-detector (light-intensity high)) 'yes)
((light-detector (light-intensity low)) 'maybe)
((heat-detector (heat-intensity high)) 'no)
```

6.3.1.2 Degrees of Belief

It is important to make a distinction between those concepts which the system is confident that the user knows, and those which are less certain, based perhaps on indirect inference or uncertain knowledge. It is less clear how fine these distinctions should be. There may be some advantages in using a numeric figure giving a degree of belief (varying from, say, -5 to 5) depending on whether the concept is definitely believed unknown, definitely believed known, or something in between. Inference rules too could have certainty factors attached to them, and beliefs updated by some numeric algorithm such as used in many expert systems. Yet these calculations tend to be ad hoc, based on no clear theory. Furthermore, it is not at all clear whether these distinctions are worth making.

The EDGE system therefore uses just the four values which are needed for making discourse decisions. These are:

- 'Yes': The system is fairly certain the user knows the concept, based on direct information from the user or fairly certain inferences (such as all the sub-skills being known).
- 'Maybe': The system has some evidence that the user knows the concept, such as by indirect inference or by the system telling the user something.

- ‘Don’t know’: The system has no information. No inferences may be made.
- ‘Probably not’: The system has some evidence that the user does not know the concept. A further value ‘No’ could be used for when the system has direct negative evidence, but this distinction has no effect on the discourse in the current system.

6.3.1.3 Direct and Indirect Inferences

The system uses a range of direct and indirect inferences to determine how strongly a concept is believed known. The direct inferences are based on direct interactions with the user, while the indirect inferences use the difficulty levels and relations between concepts in the domain. Figure 6-1. illustrates the direct inferences used, while figure 6-2 illustrates the indirect inferences⁴. These rules are considered in the order given, so that more certain inferences are used first. This avoids consideration of conflicting information, aiming only at a ‘best guess’ at what the user knows based on the most certain current information. However, it is possible that examining all sources of information, including direct inferences made in the past, would improve performance.

Although there is a certain amount of arbitrariness about the ordering of the rules, it should be clear that a lot of information may be obtained from the discourse without asking the user direct questions about what they know. The information may be very uncertain, but is still useful for directing the discourse.

This second set of rules is only used when the system needs to know whether the user knows a concept, rather than when new evidence is obtained. This

⁴In practice (in the current system) it is not always possible to deduce that the user definitely understands a proposition — there may be no appropriate question to ask — so the system may conclude that a concept is known if at least some sub-concepts are known and the others maybe known.

Dialogue Exchange	Inference
System tells user X, User acknowledges	User maybe knows X
System asks user X, User replies correctly	User knows X
System asks user X, User replies incorrectly	User doesn't know X
System asks user if she knows X	User's reply
User asks X	User doesn't know X

Figure 6–1: Direct Rules for Inferring What User Knows.

Condition	Inference
All subconcepts known/unknown	Parent concept known/unknown
All subconcepts known or maybe known	Parent concept maybe known
Concept difficulty greater than level of expertise of user	Concept probably unknown
Concept difficulty less than level of expertise of user	Concept maybe known
Parent concept known/maybe known	Subconcepts maybe known

Figure 6–2: Indirect Rules for Inferring What User Knows

avoids the need for complex truth maintenance schemes, but may lead to inefficiency.

6.3.1.4 Levels of Expertise

Most of the inference rules require no discussion. However, the use of difficulty levels⁵ and expertise is a little more complex.

In the EDGE electronics explainer, four levels of expertise are defined. These are:

Novice: Knows no electronics and virtually no electricity.

Beginner: Remembers electricity from school, but no electronics.

⁵The difficulty levels, strictly speaking, define whether the concept should be known by a user at a certain level, rather than the actual difficulties of the concepts.

Intermediate: Knows a little electronics.

Expert: Knows a fair bit of electronics.

These are given numerical values (1-4) for convenience. Difficulty levels may be defined for any of the concepts in the domain. For example, a transistor's behaviour might be familiar to someone who knows a little electronics, so has difficulty level 3. Most features of a resistor will have difficulty level 2, while how an astable multivibrator works will have difficulty level 4. If the user is of intermediate level of expertise it is therefore likely that she will know all about resistors, and unlikely that she will know how an astable works. However, it is less sure whether she will know what a transistor does.

Difficulty levels of other concepts may be inferred using defaults. For example, suppose the following domain plan is defined:

```
(defplan what-it-does (device)
  :subgoals ((foreach input in (getslot device 'inputs)
    (in/out (device input
      (get-output device input))))))
  :difficulty (3 (light-detector) 2 (NAND) 4))
```

The first value in the difficulty level set defines a default to be used for all the devices (arguments) not explicitly mentioned. So the difficulty of 'what the heat detector does' would be 3, and the difficulty of 'what the light detector does' would be 2.⁶

These default rules introduce even more uncertainty into the system. Yet such difficulty levels are only ever used to make initial assumptions about the user's knowledge (the user 'maybe' knowing the concept), so there is no inconsistency

⁶Originally difficulty level defaults also used sub-skill relationships — however, this appears to be subsumed by the basic user model indirect inference rules and to be unnecessary.

introduced. The defaults are there to make it simpler to provide user modelling information given an existing set of discourse content plans without having to explicitly set difficulty levels for every plan and possible set of arguments.

6.3.1.5 Updating the Level of Expertise

As the discourse progresses, it may be necessary to revise the user's 'level of expertise', used in making many of these inferences. For example, if the user correctly answers a number of 'difficult' questions, or asks about a number of 'easy' concepts, then this suggests that the level of expertise given is incorrect. The EDGE system therefore accumulates an average rating of concept difficulty (based on a total rating divided by the number of updates) which is used to update the level of expertise. This total rating is revised as follows:

- When the user correctly answers a question, then the difficulty value of the associated concept is added to the rating.
- When the user incorrectly answers a question, or asks a question, then one less than the difficulty level of the concept is added to the rating.

Initially the expertise level may be given with a certain weighting, and this is used to bias the initial ratings. For example, suppose the expertise level was 2, with weighting 5. After a user correctly answers a question of difficulty 3, then the rating would be $(2*5+3)/6$, with a weighting of 6. If a user then incorrectly answers a question of difficulty 2, then the new rating would be $(2*5+3+1)/7$. The increased weighting represents the increased evidence for the stereotype, and gives the system inertia while allowing it to revise that stereotype. The expertise level (stereotype) used is the expertise rating rounded to the nearest integer — this avoids the value being used when there is very marginal information.

6.3.1.6 Issues and Extensions

It has been shown above how a great deal of information may be obtained about the user's knowledge from a small number of exchanges with the user. However, there are various extensions which could be made to the system. If the user was allowed to ask questions and give information in unconstrained natural language (rather than from menus) then more information could be gained. For example, the system could assume that the user knew all the presuppositions of a question [Chin 88] or use the rules of communication discussed in [Kass & Finin 87].

Another issue is the role of the prerequisite and sub-skill relations between concepts in the domain. These are really curriculum relations rather than domain relations. They do not include particular relations between objects in the domain model such as *specialisation* or *part of*, or even relations between events being described, such as *causes* or *precedes*. Yet these relations too could be used to make inferences about the user's knowledge. For example, if the user knows the behaviour of a particular type of NAND gate, it is reasonable to suppose that she knows the behaviour of all NAND gates if she can identify them as such, though may only know how one of them works. To utilise these sorts of inference rules it would be necessary to link the user model (currently based on the explanation content plans) back to the underlying domain knowledge. Then inference rules could be devised which acted on the underlying domain knowledge relations as well as the curriculum relations implicit in the plans. This distinction between domain relations and curriculum relations in user modelling does not seem to have been made by Kass or Sleeman.

A related issue concerning domain knowledge relations is the question of how users infer and learn new knowledge. It can be expected that the user will be able to at least make many of the inferences that the system can. For example, if the user knows that a light detector circuit is a potential divider circuit, and potential divider circuits tend to be used to vary a voltage given a variable resistance, then the user should be able to deduce a number of things, including

that the light detector has a variable resistor, and that its function is to give a variable output voltage. Perhaps the user modelling system should assume that the user makes such inferences — yet people don't make all the inferences possible, so there would need to be a theory of human plausible reasoning. To go even further, the system should deduce what the user might learn spontaneously, by analogy or generalisation for example. For this we'd need a computational theory of human learning.

These are just some of the issues which are being considered by work on user modelling today. Others include the misconceptions people have in various domains — how they come to have these misconceptions, how to recognise them, and what to do about them. If the primary aim of the EDGE system was to provide an environment for effective teaching in a particular domain, then the problems of representing and challenging misconceptions would have to be addressed. However, for a discourse model these issues are of secondary importance. What is important is how some information about the user's knowledge may be inferred in the process of a natural discourse, and how these in turn affect the discourse. The EDGE system demonstrates this.

6.3.2 Effect on Discourse

The model of the user's knowledge affects the discourse in many ways. These have been introduced in the previous two chapters, but will be summarised here.

6.3.2.1 Effect on Discourse Content

The user model may affect the content of the discourse in two main ways. First, it may affect the strategies used to teach a concept, using constraints on the content plans. This could include selecting different models to use, different task decompositions or choosing to describe using an analogy. Second, it may affect what is included in the discourse, using preconditions on the content plans. For

example, the behaviour of components may only be described if not known by the user.

6.3.2.2 Effect on Dialogue

Depending on whether a fact is believed known, maybe known, or probably unknown, different types of exchanges may be selected. The aim of the system is to increase its certainty that the user knows the fact. If the fact is believed unknown, this is best realised by telling the user the new fact. If it is believed maybe known, this is best realised by asking the user a question. If it is believed known then the fact should still be told — decisions about what to leave out are the responsibility of the content planner.

The other effect on the dialogue is when the system needs to know whether the user knows some (possibly complex) concept, and has no information about it. This leads to an exchange where the system asks the user directly whether she knows the concept.

These effects are influenced by the goal of building up a model of what the user knows, as well as by the goal of effectively teaching the user.

6.3.3 Example

Full annotated examples of explanatory dialogues are given in the next chapter. However, it is worth discussing a short example. The following is the discourse resulting from the goal of explaining what a light-dependent resistor does, given no initial information about whether it is already known:

Do you know what a light-dependent resistor does given different inputs?

— I think so

Well, we'd better go through it to check.

What do you think its output resistance is when the input light intensity is high?

— **High**

No, not quite, it's actually quite low.

What do you think its output resistance is when the input light intensity is low?

— **Quite high**

Yes, excellent, it's quite high.

This short example illustrates how the user model influences, and is influenced by the discourse. It is annotated in detail in figure 6-3 to show the reasoning process.

There are many possible problems with the inferences illustrated. For example, it could be argued that once the user has got the question relating to one sub-concept wrong, the system should assume that the user does not know the other sub-concept. Or that correcting the user after an incorrect answer should not lead to the assumption that the user now 'maybe' knows the correct answer. Also, assuming that a concept is known if its sub-concepts are known or maybe known may be too strong an inference⁷. The rules and preferences used in the current system are not definitive, and should be revised following further empirical evaluation. The important thing is that an increasingly detailed model of the user's knowledge may be built up and used without destroying the coherence of the discourse.

6.3.4 Other Types of Discourse

The above approach has to be modified slightly for different discourse types. The structuring of the content and the allowed exchanges may both be different.

⁷There is perhaps also a more fundamental problem concerning how far users use intelligent guesswork based on the past explanation in giving answers. A novice user may correctly answer questions based on little understanding.

Initially the system needs to know whether the user knows what a light-dependent resistor does. Unusually it has no information available, as the expertise level of the user is the same as the difficulty level of the concept, so it asks a direct question.

Do you know what a light-dependent resistor does given different inputs?

— **I think so.**

The user's answer causes the abstract concept (what an LDR does) to be explicitly marked as 'Maybe' known. As it is not definitely known the system's response is to go through it anyway.

Well, we'd better go through it to check.

The subgoal of explaining what it does for a particular input is initiated, which leads to a teaching exchange. Selecting the appropriate exchange requires information about whether the user knows the associated concept. As there is no information about whether the sub-concept is known, the system uses the knowledge about whether the parent concept is known (Maybe) and so chooses to ask a question.

What do you think its output resistance is when the input light intensity is high?

— **High**

No, not quite, it's actually quite low.

The user answers incorrectly, so momentarily the system marks the sub-concept as unknown. However, the system corrects the user, so it marks the sub-concept as maybe known. Again using the parent concept's value as default, the system asks a second question:

What do you think its output resistance is when the input light intensity is low?

— **Quite high**

Yes, excellent, it's quite high.

Now the sub-concept (what the LDR does when the light intensity is low) is marked as known. The other sub-concept is still marked 'maybe' known, so the system would infer that the user now knows the parent concept (what the LDR does).

Figure 6-3: Annotated User Modelling Example

6.3.4.1 Conceptual Structure

In all the discourse types considered, there is some hierarchical structure to the discourse content which relates to the conceptual structure of the domain. However, it may not always be appropriate to use only the simple sub-skill relations assumed in the preceding discussion. For example, the relations in justifications or in primarily causal explanations may require different types of inference rules.

For task explanations there should be knowledge both about whether the user knows how to execute a task, and whether it has been executed. However, the basic user modelling inference rules are similar.

6.3.4.2 Roles and Exchange Types

As discussed in the previous chapter, different exchange types are allowed depending on the relative roles of speaker and hearer. In some situations, where the speaker has a very dominant role, the hearer may only be allowed to answer exchanges, and never initiate them. In the EDGE model of classroom discourse students are allowed to initiate questions, but not to initiate transactions (change the topic) or volunteer information. The exchanges allowed each participant are still unequal due to the unbalanced social roles.

In other types of discourse, where different exchanges are allowed, the sources of user modelling information will change. For example, where the user may volunteer information (as in ordinary conversation) then inferences must be made from what she says (cf. [Kass & Finin 87]). If the discourse type does not allow the system to ask direct questions of the user, then more certain information about the user's knowledge must be gained in other ways.

In the cookie explainer, the system may have the role of an advisor, rather than a tutor. When the user asks a help system for an explanation it would not make sense for the system to answer:

Well, you try and tell me how to do it.

yet this would be a very reasonable response in a teaching situation. In a help system, on the other hand, it is quite reasonable for it to ask questions about the user's knowledge and ability such as:

Do you know how to roll out the cookies?

Therefore, in the cookie explainer, if the system believes the user 'maybe' knows how to do a task, it will ask such a direct question. As the user is more likely to know what she knows in this context than in tutorial discourse, this provides fairly certain information about what the user knows how to do.

6.4 Conclusion

The EDGE user modelling scheme is an extended overlay model similar in many ways to the UMFE system [Sleeman 85]. It allows assumptions about the user's knowledge to be updated as an explanatory dialogue progresses⁸. All the criteria presented in section 6.2 are satisfied:

Direct Inference Rules: These are used to obtain information about the user's knowledge from all the different types of interaction in the discourse system.

Indirect Inference Rules: The structure of the domain knowledge and the difficulties of concepts are used to indirectly infer what the user may know.

⁸It is important to note that these are just assumptions. An interactive explanation system does not demand correct knowledge about the user, but approximate knowledge may improve its performance.

Changing Assumptions: Indirect inference rules are only applied when needed, avoiding problems of changing assumptions about the user's knowledge.

Using and Inferring Model: The application system (the discourse planner) and the user modelling system are integrally related.

Representation: Concepts may be represented at different levels of detail, and believed understood to various degrees.

The main contribution of the approach is the way it is integrated with the application (discourse) system. A wide variety of inference rules are used based on the different types of interaction in the system. All the user model information required may be obtained in the context of a coherent discourse, though the goal of user modelling may influence the discourse. The user model representation is based on existing conceptual structures in the discourse (content) planner, allowing concepts to be represented at various levels of abstraction — those levels needed by the application system.

The main limitations of the approach are related to the limitations of overlay modelling in general. Misconceptions are not explicitly represented (though they may be implicit in the remediation strategies), and the system assumes that the user's relevant knowledge is a subset of the system's. It further assumes that the user's goal is to increase her knowledge in line with the system's. Neither of these assumptions are supported by current pedagogical thinking, but provide a useful basis for generating explanatory discourse in many contexts.

Chapter 7

Application and Examples: How Circuits Work

The previous three chapters have shown how general principles have been used in developing the EDGE model of explanatory discourse. This model is applicable to a range of domains and to a range of discourse types. However, the main effort has gone into developing a working computational system to explain how simple electronic systems work. This domain is sufficiently complex to provide a real test of the principles involved.

In order to plan explanations in a new domain you need a good understanding of the conceptual structure of the domain, and of how 'expert' explainers use that knowledge to generate an appropriate explanation for a particular student. One source of such information is an analysis of expert explanations, as described in chapter 3¹. This chapter will show how the explanation structures inferred from those analyses have been used to plan model-based explanations of how circuits

¹Analyses should ideally also include psychological/educational studies of how people learn in the chosen domain. There was not scope in this research to include this type of analysis (and there appears to be little existing research in the particular domain considered). The emphasis in this thesis is therefore on studies of human expert explanations.

work. The chapter will include extensive annotated examples of explanations generated by the system.

7.1 Review: Explaining How Things Work

Much of the work on generating ‘qualitative’ explanations of how physical systems work has concentrated on the underlying device models and the reasoning processes involved in working out device behaviour. De Kleer, in his early work on qualitative reasoning, showed how qualitative models of circuits could be developed which could be used with a ‘qualitative reasoning system’ to generate plausible explanations of a circuit’s behaviour [deKleer 84]. For simple examples these corresponded fairly well with expert causal explanations of circuit behaviour.

At about the same time, Forbus and Stevens were using similar techniques to generate explanations which combined a qualitative causal description with a graphical simulation [Forbus & Stevens 81] — an idea also developed by Simmons [Simmons 85], in the Quest system [White & Frederiksen 85, Frederiksen & White 88] and more extensively in a reworking of the ‘STEAMER’ intelligent tutoring system using qualitative models [Falkenhainer & Forbus 88]. Both Forbus and de Kleer were motivated by an initial interest in intelligent tutoring systems, but the field of qualitative reasoning has subsequently become dominated by more theoretical work on developing such systems.

The above work concentrated on qualitative, causal explanations of how something worked. However, related work at the same time by Stevens and Steinberg (also in the ‘STEAMER’ project) showed that there are a wide range of types of explanations of physical systems [Stevens & Steinberg 81]. They classified a large number of explanations in Navy training manuals and came up with a typology of explanation types observed. They describe nine types of explanations concerned with physical system behaviour, summarised below:

Behavioural: The input-output behaviour of the system.

Physical-causal: A causal explanation based on the physical decomposition of the system.

Synchronous: Used to discuss equilibrium states etc.

Information flow: Sensing and propagating information in a system.

Stuff-state-attribute: Explanations which describe state or attribute changes.

Stuff-as-a-transport-medium: Explanations describing how some 'stuff' (e.g., water) is used to transport something such as energy.

Componential: Describing the components of the system.

Topological: Describing components and connections.

Geometric: A more quantitative description of structure.

Stevens and Steinberg suggest that this enumeration of explanation types may be based on four main distinctions. These are *structure vs. mechanism*, *individuated vs. aggregate events or components* (e.g., using prototypical instances and averages), *qualitative vs. quantitative* and *external vs. internal properties*. They acknowledge that in practice a combination of more than one type of explanation may be appropriate, suggesting that instructional goals might include things like: "To describe a reducing valve, use one physical-causal explanation at the level of the first set of sub-components, and one behavioural explanation; but to describe a venturi, use one aggregate objectifying individual molecular events and one stuff-state-attribute with energy objectified."

At this point Stevens and Steinberg's work begins to sound like the *independent* work on text generation by McKeown and Paris (described in chapter 2). Both classify types of explanation or utterance and show how a complete description

may be structured from several types. Stevens and Steinberg give a more detailed classification of explanation types, and suggest that different types of system will need different types of explanations. McKeown classifies single utterances in a more domain independent way, but attempts to obtain general schemas which may be used to generate descriptions in a wide range of domains.

It is clear from this work that there is more to explaining how things work than just giving a qualitative causal explanation. Explanations are complex things involving structural and functional information as well as causal, and causal explanations may be of many different types. Hollan and Hutchins go further and suggest that qualitative models are inadequate for explanation not just because of the restricted 'type' of explanation that they apply to, but because of the ambiguities and limitations inherent in such reasoning systems [Hollan & Hutchins 84].

Human explanations, as well as including a wide range of 'types' of utterance or description, also depend on the knowledge of the hearer. Paris shows that explanations for novices include mainly a causal description, while explanations for experts include mainly a structural description of a system [Paris 87]. Cox, on the other hand, claims that people go through a number of stages in understanding complex physical systems, and suggests that explanations should reflect these levels of understanding [Cox *et al* 88]. The first of these are the stage of *figurative knowledge* involving the attributes and identity of the object; the *functional* stage involving the function and behaviour of the object; and the *cause-effect* stage involving causal relationships. It seems likely that these stages come before Paris's expert level — only when the hearer has a good understanding of a system is a simple structural or componential description adequate.

Miyake shows that for a complex system — in this case the workings of a sewing machine — it is insufficient to claim that people first learn the function, then the mechanism [Miyake 86]. In his experiments the process is iterative — through a process of negotiation and explanation people went through six levels

of a *function-mechanism* hierarchy. At each stage they seek to understand a mechanism assumed at the previous level.

It appears then, that the process of understanding how something works is far from simple. Explanations should reflect that process, and not be limited to a causal trace of device behaviour. They may consider the multiple mental models used in understanding ([Collins & Gentner 83, Moyse 89]), stages of understanding ([Miyake 86, Cox *et al* 88]), and how different types of explanation or utterance are combined depending on the system explained and the knowledge of the novice ([Paris 87, Stevens & Steinberg 81]). The rest of this chapter follows on from the analysis in chapter 3, showing in detail how explanations are structured in the domain of simple electronic circuits. However, first the criteria which will be used in developing the plans will be summarised.

7.2 Criteria for Explaining How Things Work

From the discussion in the last section, and the empirical analysis described in chapter 3, a number of principles for explaining physical systems can be obtained. The following are the principles which will be used for the EDGE plans.

Structure of Explanations (and Explanation Types): Explanations should involve behavioral, causal, functional, structural and identification/attribute information [Stevens & Steinberg 81, McKeown 85, Cawsey 88a]. Which types of explanation are included should depend on the device being explained and the prior knowledge of the hearer [Paris 87, Stevens & Steinberg 81]. The order of explanation types is important in allowing the hearer to assimilate and understand the new information [Cox *et al* 88], and typical orderings or schemas can be observed in expert descriptions [McKeown 85] (and in chapter 3).

Value Display and Representation: Explanations should normally use qualitative values rather than numerical values, and graphical representations should be used in conjunction with textual description [Forbus & Stevens 81, Falkenhainer & Forbus 88, Simmons 85]. In some cases a very simple qualitative model is sufficient for such explanations (e.g., [White & Frederiksen 85]).

Levels of Explanation: Explanations should be given at different levels depending on the understanding of the student [Miyake 86]. It should be possible for the student to repeatedly ask 'how does that work' when given some behavioural description² — though in practice there will always be some cutoff point where such explanations are too hard. Understanding may progress through a process of questioning and negotiation.

There is really an additional criterion apparent from the work described above — that different models should be available for use in the explanations. This might include qualitative vs. quantitative, voltage vs. current, or system vs. equation based models [Gentner & Stevens 83, Moyse 89]. However, this issue is beyond the scope of the current research.

Based on these principles, explanation plans and device models have been developed. The next section will show how these plans and models have been used to generate explanations of circuit behaviour.

²Sable et al. have used this idea when generating explanation of how to sail between two points [Sable *et al* 85]. The explanation is given at a high level of detail, and more detailed plans can be obtained as requested.

7.3 The EDGE Circuit Explainer

Generating explanations in a new domain involves writing a set of skeletal plans for explaining in that domain, and deciding on a representation for the underlying knowledge in that domain. This section will therefore discuss the plans and the models developed for circuit explanations, before giving a number of detailed example explanations and discussing how the approach may be generalised to other domains.

7.3.1 Plans to Explain Circuits

First, the explanation plans used to explain how circuits work will be discussed in detail. The actual plans used are given in appendix B. The examples given below may in some cases be slightly simplified.

7.3.1.1 Structuring the Explanation

An explanation is structured so that understanding device structure is viewed as a prerequisite to understanding behaviour, and understanding device function and identity are prerequisites to understanding the device structure. In other words, first a link must be made into the student's existing knowledge (identity), then its function should be given to motivate interest in the device. Next the structure is given, which is an obvious prerequisite to understanding a causal explanation. An explanation of how a circuit works consists of a causal description and a summary of behaviour (as suggested by Stevens and Steinberg for some systems). Figure 7-1 shows the plans used to achieve this high level structuring³.

³Note the use of *sub-transactions* to structure the discourse. These will cause appropriate discourse markers and/or meta-comments to be added to discourse segments. The

```

(defplan how-it-works (device)
  :constraints ((getslot device 'structure))
  :preconditions ((structure (device)))
  :subgoals ((interaction sub.transaction
    ((plan 'teach 'causal-behaviour (list device))
      'mid))
    (interaction sub.transaction
      ((plan 'teach 'what-it-does (list device))
        'conclusion)))
  :template ("how" (deviceref device) " works")
  :difficulty (3 light-detector 2 low-heat-warning 4))

(defplan structure (device)
  :preconditions ((what-sort-of-device (device))
    (what-its-for (device)))
  :subgoals ((teach components (device)))
  :template ("what" (deviceref device) "is like"))

```

Figure 7-1: High Level Plans to Explain How Circuits Work

In the existing system the *structural* description simply involves describing the components of the device — the rest is clear from the diagram which is referred to in this description. The *what-sort-of-device* plan will look for a link with existing knowledge — some X such that the device is an X and the hearer is familiar with X. The *what-its-for* plan will describe how the output value depends on the input value(s).

From these plans it should be clear how an explanation is structured to include different types of utterance, and how the content will depend on the hearer's knowledge. If the hearer is familiar with the 'structure' of the device (necessitating, in this model, being familiar with the identity and function of

plan function constructs a representation of the given plan allowing pointers to parent and sub-plans.

the device), then a simple causal explanation will be given. If the hearer is not familiar with the device then the whole series of explanation types will be given — identity, function, structure, causal and behaviour. If the hearer has intermediate knowledge — such as being familiar with the device but not with one of its component behaviours — then this behaviour is given before the causal explanation.

7.3.1.2 Component Behaviours

A *structural* description (perhaps better called a *componential* description) includes a description of the identity and behaviour of each component of the device which is not already understood⁴. This is achieved using iterative and conditional statements in the appropriate plans. These are given in figure 7-2.

There are alternative ways of structuring the plans which will achieve very similar results — perhaps in figure 7-1 the type and function of a device should be prerequisites of the *how-it-works* plan and not the *structure* plan. The plans shown here are just presented as one way of achieving the sort of structured, tailored explanation wanted given the constraints from the empirical analysis and from the literature.

⁴Note that there appears to be some problem with high level prerequisite goals. If a goal is a prerequisite then its subgoals should generally only be executed if they are not understood, while if it is a sub-goal of a plan then all its subgoals should normally be executed so that the explanation is coherent. This might in general require interpreting such goals differently, though in the current system conditionals are used to state explicitly when subgoals should be used depending on the hearer's knowledge.

```
(defplan components (device)
  :constraints ((getcomps device)) ;; model includes components
  :subgoals ((foreach comp in (getcomps device)
    (if (greaterp (importancelevel
      (list comp) 'what-it-does)
        1)
      (teach component (comp))))))
  :template ("what" (deviceref device)
    "s components are like."))

(defplan component (device)
  :preconditions ((what-it-looks-like (device)))
  :subgoals ((if (not (understands
    (list device) 'what-it-does))
    (interaction sub.transaction
      ((plan 'teach 'what-it-does (list device))
        'sub))))))
  :template ("what" (deviceref device) "is like"))
```

Figure 7-2: Conditional and Iterative Statements in Component Plans

```

(defplan what-it-looks-like (device)
  :subgoals ((call pointat (device))
             (interaction teaching.exchange
              ((list 'describe-comp device))))
  :template ("what" (deviceref device) "looks like.))

(defplan in/out (device input output)
  :subgoals ((call putresult (input (getslot device 'inwindow)))
             (interaction teaching.exchange
              ((list 'describe-event device input output)))
             (call putresult (output
                              (getslot device 'outwindow))))
  :template ("what the output of" (deviceref device)
             "is when its inputs are:" inputs))

```

Figure 7-3: Graphical Actions in Plans

7.3.1.3 Graphical Actions

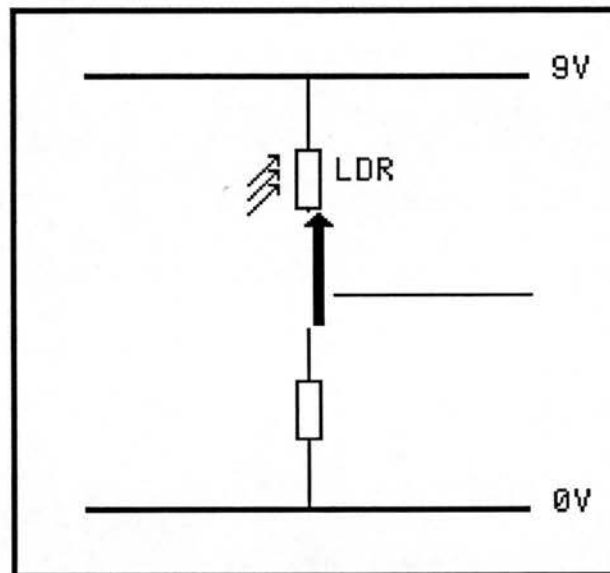
Plans may also involve physical (graphical) actions. These are currently implemented as Lisp calls. The plans in figure 7-3 illustrate pointing and displaying qualitative values on meters⁵, while figure 7-4 illustrates the actual output of the system for the light detector circuit when these two plans are called.

Note that the 'devices' passed as arguments in these plans are particular instances, such as a particular light-dependent resistor. These instances have input and output meters (windows) where their input and output values may be displayed. The underlying device models will be discussed in section 7.3.2.

The current formalism allows graphics actions at any point in a plan — however, in human discourse gestures are coordinated at a finer level of detail

⁵These plans are simplified to deal with one input systems — The plans given in appendix B deal with more complex systems with several inputs.

“This component here is a light dependent resistor.”



“When the input light intensity (L1) of the light dependent resistor is high (light) the output resistance (R1) will be quite low.”

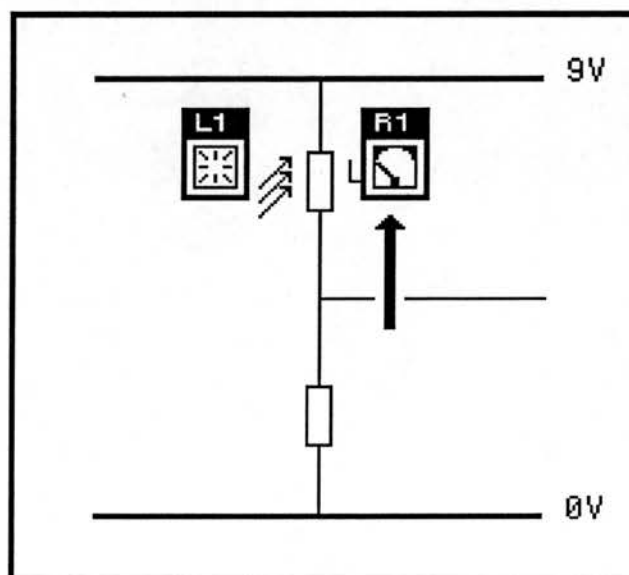


Figure 7–4: Graphical and Textual Output in Circuit Explanations

corresponding to phrases or tone units which may come within a teaching exchange [Kendon 83]. It is therefore more likely that graphical actions should be defined as part of an utterance rather than as part of a discourse content plan.

7.3.1.4 Causal Behaviour

The most important part of an explanation of how something works is arguably the description of the causal processes involved. This is the aspect which work such as [Falkenhainer & Forbus 88] and [deKleer 84] has concentrated on. This thesis deliberately circumvents some of these problems by concentrating on a type of electronic circuit where the behaviour is very simple. The behaviour of these systems can be explained as a function of the input-output behaviours of the components — no feedback is involved and each device has a single significant output value. Larger circuits may be viewed as composed of a number of simpler circuits with given input-output behaviour⁶. However, the approach to planning explanations will generalise to more complex systems. A plan to explain the causal behaviour of a device given some inputs will pass control to an interpreter whose job it is to generate the causal steps of the explanation at the highest level of detail. The relevant plan is given in figure 7-5⁷. The interpreter, on obtaining a causal step, will pass control back to the explanation planner, which may choose to convey that step or perhaps to obtain a more detailed explanation⁸.

⁶This approach is taken from a recent electronics textbook: [Johnson 87].

⁷Note that these plans give the causal trace for each significant input. However, the user model could be used so that the system only explains input-output behaviours that are not already known.

⁸In the current implementation this decision is made within a function called by the interpreter. A plan to give the input-output behaviour of the component may be put on the stack, or a plan to give a more detailed causal explanation of its behaviour.

```

(defplan causal-behaviour (device)
  :constraints ((getslot device 'structure))
  :subgoals ((foreach input in (getslot device 'inputs)
    (teach particular-causal-behaviour
      (device input))))
  :template ("how" (deviceref device)
    "works given different inputs."))

(defplan particular-causal-behaviour (device input)
  :subgoals ((interaction teaching.exchange
    (list 'suppose device input))
    (call put-result (input (getslot device 'inwindow)))
    (teach causal-bit (device input))))
  :template ("how" (deviceref device)
    "works given the following inputs:" inputs))

(defplan causal-bit (device inputs)
  :subgoals ((call run (device inputs))))
;; pass control to interpreter

```

Figure 7-5: Plans to Explain Causal Behaviour

In the current system the interpreter is fairly simple, but in general it might be a reasoning system such as described in [Falkenhainer & Forbus 88].

When the system describes a high level step in the causal explanation, the student always has the opportunity to ask for a more detailed explanation. Thus the explanation of a complex system may progress through interaction with the student.

The causal explanations involve *qualitative* values. Because of the simplicity of the example circuits it is not necessary to do complex reasoning about intervals. The following values/intervals are used:

Low: 0 V, 0 Ohms etc.

Quite Low: The interval (0, 4.5 V) or (0, 1 KOhm).

Average: 4.5V, 1KOhm.

Quite High: (4.5, 9 V) or (1, Inf KOhm).

High: 9 V or Infinite resistance.

These were the sorts of values used in human explanations. The associated intervals are only given to show that the underlying reasoning is still sound. For example, if a potential divider has one quite high resistance one quite low one (near the 9V rail) the output voltage will be quite high. If it has one quite high one and one low one the output voltage will be high. The ‘average’ resistance of 1KOhm is defined arbitrarily.

7.3.1.5 Alternative Plans

In any explanation system there must be different ways to explain the same thing. This might involve using different models or different explanation strategies. When there are alternative strategies, then not only can an explanation be better tailored to the particular student, but when an explanation fails (the student doesn’t understand) there is more potential for repair — explaining it another way, rather than just giving more detail [Moore & Swartout 88b].

This thesis does not address the problem of re-explaining when things go wrong, though this would be a straightforward and important extension, discussed in chapter 8. However, alternative plans are available so that more appropriate explanations may be generated initially. There are two main cases where there is a choice. First, there is an alternative plan for explaining the structure/components of a device using an analogy with a similar known device. Second, steps in a causal explanation may be given as *black box* behaviours, or explained in more detail. Both these plans also prove useful in remediation strategies, when the student answers a question incorrectly.

Using the plans described, it should be clear how explanations are structured for a particular student, and how they meet the criteria given at the beginning of this chapter. However, before moving on to give examples of the system explaining a range of circuits, the underlying device models will be described.

7.3.2 Device Models, Instances and Diagrams

The plans discussed above take particular devices as arguments. These are instances of generic device types, and inherit properties from the generic object. For example, a plan to describe a component may have a particular light-dependent resistor as its main argument (say LDR6). This will have a position (so it can be pointed at), input and output meters which are used to indicate values, and when appropriate, pointers to component instances. It will inherit its other properties from the general light-dependent resistor definition

At the beginning of an explanation, if the device named has not been explained before, the system will create a diagram from the structural description of the device and routines for drawing components. At the same time it will create object instances for the device, its sub-sections (if there are any) and all its components. This is managed so that input and output meters correspond, so a device which takes its input voltage from the output voltage of another device will share that meter. The new structural description based on the component instances is used as the new structure property of the device instance, so that when components are described or a causal description given using that description, all the appropriate component instances are used. The construction of the diagram also causes the different regions of the graphics window to have attached questions menus. This follows from the observation in chapter 3 that many questions were asked in the context of the diagram by first pointing at the relevant region. Using the *active-regions* of the diagram it is possible to point and click on a component or circuit section and get a menu of questions which may be asked about that section.

```
(defmodel heat-detector
  :name "heat detector circuit"
  :inputs ((heat-intensity high) (heat-intensity low))
  :circuit-type potential-divider
  :device-type "input transducer"
  :behaviour run-heat-detector
  :structure (lambda (heat)
    (potential-divider
      (fixed-resistor 'medium) (thermistor heat))))
```

Figure 7-6: Defining a Heat Detector Circuit

Before a new device type may be explained, it is necessary to define its structure, type, typical inputs, black box behaviour and so on. This is done using definitions such as the one in figure 7-6. The black box function *run-heat-detector* must then be defined. Larger circuits may then be built up of smaller ones, defining the structure as a function of circuit sections⁹.

This representation could be improved in many ways, allowing a richer hierarchy of circuit types or improving the *structural* description to allow a more realistic range of circuits. This section serves simply to show that a complex graphical-textual interactive explanation may be based on a very simple domain model.

7.3.3 Example Explanations

This section will go through a number of detailed examples of explanatory dialogues generated in this domain, showing how the user model, content plans and discourse plans are used. In each case the dialogues are generated by the system

⁹See [Clancey 88] for a discussion of functional and causal models in tutoring.

with the author acting as student. Some of the dialogues will be displayed in two columns. The first will display the goals or plans being executed (a trace generated by the system). The second will include the actual dialogue in quotes, and comments on the inferences being made in italics.

The aim of this section is to give an overview of how a complete explanation is generated. Examples of how the system deals with interruptions and how it responds to student errors are given in chapter 5.

7.3.3.1 Explaining How a Light Detector Works

The first detailed example is an explanation of how a *light detector* circuit works. Initially the user model is set so that the system believes:

- The user knows what a light detector circuit is.
- She may or may not know what it is for.
- She doesn't know everything about its structure and components.
- In particular, she doesn't know exactly what a light-dependent resistor does — or at least, only maybe knows what it does for a high input light intensity.
- She only maybe knows what a potential divider circuit does.

Figure 7-7 shows the complete resulting dialogue, given an initial call to give an *informing transaction* of how the light detector works¹⁰. A large amount of

¹⁰As in the other examples the English has been copied from the program output, replacing menu calls with text (and occasionally correcting case etc where a value or template has been incorrectly defined).

OK, we're going to be explaining how the light detector unit works.

Do you know what it is for?

– No

It's purpose is to produce an output voltage which depends on the input light intensity.

This component here is a light-dependent resistor. *Points at LDR.*

Now, let's go through what the light-dependent resistor does.

What's the output resistance going to be when the input light intensity (L1) is high (light) ?

– Quite low.

Correct, excellent, it's quite low.

When it is low (dark) the output resistance (R1) will be quite high.

Right, we better explain the behaviour of the light detector unit.

Suppose the input light intensity is high (light)

When the input light intensity (L1) is high (light) the output resistance (R1) will be quite low.

What's the output voltage (V1) of the potential divider going to be when there is a quite low input resistance (R1) and a average input resistance (R2).

– Quite high

Yes, great, it's quite high (towards 9V).

Suppose the input light intensity is low (dark).

When the input light intensity (L1) is low (dark) the output resistance (R1) will be quite high.

What's the output voltage (V1) of the potential divider going to be when there is a quite high input resistance (R1) and a average input resistance (R2).

– Low (OV)

Wrong, not quite, it's quite low (towards 0V)

So, What's the output voltage (V1) going to be when the input light intensity (L1) is high (light)?

– Quite high

Precisely, excellent, it's quite high (towards 9V).

What's the output voltage (V1) going to be when the input light intensity (L1) is low (dark)?

– Quite low

Precisely, splendid, it's quite low (towards 0V).

Is that enough about how the light detector works?

– Yes

Figure 7–7: Explaining How the Light-Detector Works

reasoning is involved in generating each utterance of the explanation. Figure 7–8 shows a trace of the goals called by the discourse and content planners in generating the introductory comment and in deciding on the first question to ask¹¹. Figure 7–9 continues this example, showing how a teaching exchange is structured, and what is inferred about the user’s knowledge. Similar reasoning takes place in planning the rest of the explanation in figure 7–7.

It is worth noting a number of things in these examples. The discourse planner has been used to generate discourse markers and meta-comments at the beginning of the topic and major sub-topics, as well as to structure the exchanges with the user. The user modelling component influences the selection of what to include in the explanation and when to ask questions. Near the beginning of the dialogue the system infers that the user knows what the LDR does — it therefore doesn’t ask any more questions about this in the remaining dialogue. If the same explanation was given again it would not give the behaviour of the LDR and would ask a question about the light detector’s function. If this question was answered correctly it would infer that the student now knew the *structure* of the light detector.

7.3.3.2 Explaining Other Circuits

Almost all the examples in this thesis have been from explanations of how this simple light detector circuit works. However, the same plans may be used to explain a wide range of circuits, and could be generalised to explain how other physical devices worked. The first example in this section will therefore go through how an inverter works (figure 7–10), while the second will show part of an explanation of how a more complex circuit works — one that will switch

¹¹Interactions will be abbreviated to ‘Int’ in this example. Some Lisp calls are omitted.

(int informing.transaction ((teach how-it-works (light-detector14))))	<i>An informing transaction starts with a boundary exchange.</i>
(int boundary.exchange ((teach) open))	<i>This is simply expanded as follows</i>
(int framing.move (open))	
(int marker.act (open))	"Now, "
(int focussing.move ((teach) open))	
(int meta-comment.act ((teach) open))	"we're going to be explaining how the light detector circuit works."
(int teaching.exchanges ((teach))	<i>Start planning content to teach how light detector works.</i>
(teach how-it-works (light-detector14))	
(if (not (understands (light-detector14) structure)) (teach structure (light-detector14)))	<i>The prerequisite goal 'Structure' is replaced with a conditional goal.</i>
(teach structure (light-detector14))	<i>As the user doesn't fully know the structure/comps, teach this.</i>
(if (not (understands (light-detector14) what-sort-of-device)) (teach what-sort-of-device (light-detector14)))	<i>But the type of device is known, so don't teach this.</i>
(if (not (understands (light-detector14) what-its-for)) (teach what-its-for (light-detector14)))	<i>This goal causes the first question to be asked: "Do you know what it's for?" and the answer: "No"</i>
(teach what-its-for (light-detector14))	<i>So the system says what it's for (fig- ure 7-9)</i>

Figure 7-8: Detailed Trace of First Few Utterances of a Dialogue

<pre> (teach what-its-for (light-detector14)) (int teaching.exchange ((purpose light-detector14 light-intensity voltage))) (int teacher-inform.exchange ((purpose light-detector14 light-intensity voltage))) (int teacher-inform.move ((purpose))) (int teacher-inform.act ((purpose))) (int pupil-acknowledge.move) (int pupil-acknowledge.act) </pre>	<p><i>Select the informing type of teaching exchange as the user fairly definitely doesn't know this.</i></p> <p><i>"Its purpose is to produce an output voltage which depends on the input light intensity."</i></p> <p><i>Student clicks on 'OK'. System now infers that the user 'may' now know what the light detectors function is. It then checks to see if any parent plans are therefore complete. The parent plan 'Structure' is not complete as the components are not taught yet, so this is not updated.</i></p>
--	--

Figure 7–9: Detailed Trace of an Informing Exchange

a light on when the light intensity on an LDR drops below a set threshold¹². This second example will illustrate the use of different levels of explanation of complex systems.

In the first example the system starts off with no knowledge of the user except that she only knows a little about electronics. It can therefore only use difficulty levels to guess whether concepts are known. Several direct questions are asked about what the user knows — these are asked at the appropriate point in the dialogue as a conditional goal is evaluated. Comments are again in *italics* and user input in **bold**.

The second example (figure 7-11) shows how a complex explanation may be given at different levels. The example includes an interruption where the student wants a more detailed explanation and the main causal explanation is given at different levels depending on the user's understanding. Again, the system starts with little knowledge of what the user knows, but gains enough information from a small number of direct questions. As with all these examples, dynamic graphical actions are important, but hard to represent in a static diagram. Figure 7-12 illustrates the display at one point in the explanation.

This example perhaps takes the very simple models used to their limits — for example, a comparator is not best described in terms of input-output behaviour, but in terms of whether one input is bigger than another preset input. However, this thesis is not concerned with developing superb qualitative models, but about planning explanations and dialogues. Also, note that at the beginning of the dialogue the system goes to great lengths to tell the user that a low heat warning device is a type of warning device! The system has no knowledge of the semantics of the device names, so can not make the required inference to prevent this.

However, when compared with the SCHOLAR dialogue given in figure 2-1 it is clear that considerable progress has been made in structuring mixed-initiative

¹²The relevant circuit diagrams and models are given in appendix E.

expository dialogues. The SCHOLAR system did not plan the content or signal topic switches and had no explicit model of dialogue structure. The result was a flat, unstructured dialogue where it is hard to see what the system is trying to teach. The EDGE approach, on the other hand, bases the dialogue on explicit goals of what to teach and uses the discourse structure to deal more effectively with interruptions.

7.3.4 Explaining Other Things

This chapter has concentrated on showing how explanation plans can be developed to generate explanations in a particular domain — explaining electronic systems. The plans are based partly on an empirical analysis of explanations in that domain, and partly on related research on both explanation structure and on how people learn how physical systems work. So, how far do these plans generalise, and what is involved in generating explanations in other domains?

Most of the plan definitions are not specific to electrical circuits. Very similar plans are used in explaining and describing a wide range of physical system — Paris's schemas, for example, are very similar [Paris 87]. The next step in generalising the results would be to extend the plans so that constraints might access the type of domain model (e.g., electrical circuit, mechanical device). Using a specialisation hierarchy of types of physical system, and different specialised versions of plans and sub-plans for different types, it would be possible to economically represent explanation plans for a wide range of systems. Similar sorts of plan specialisation would be possible using a single set of discourse plans to apply across a range of discourse genre — again the type of discourse genre could

OK, we're going to be explaining how the inverter works.

Do you know what a inverter is like?

The difficulty levels for what an inverter is like and what a potential divider circuit is are set to the novice electronics level, so direct questions are asked.

– No

Do you know what a potential divider circuit is?

– Yes

A inverter is a potential divider circuit.

This component here is a transistor.

The system assumes that the user probably doesn't know that much about transistors, as they are quite difficult.

Now, let's go through what the transistor does.

When the input voltage (V1) is high (9V) the output resistance (R2) will be low (fully conducting).

When it is low (0V) the output resistance (R2) will be high (not conducting).

The system now infers that the user may understand what transistor does.

OK, let's explain the behaviour of the inverter.

Suppose the input voltage is high (9V).

What's the output resistance (R1) going to be when the input voltage (V1) is high?

– High

No, that's not right, it's low (conducting).

System now infers that student doesn't know the transistors behaviour for high input voltage.

What's the output voltage (V2) of the potential divider going to be when there is a average (few KOhms) input resistance (R1) and a low (conducting) input resistance (R2)?

– Low

Correct, splendid, it's low (0V).

System now infers the student does know what a potential divider does given these two resistances. The dialogue then continues in similar manner to the one in figure 7-7

Figure 7–10: Explaining the Inverter Given Uncertain Knowledge About User

OK, we're going to be explaining the behaviour of the low heat warning device.
Do you know what a low heat warning device is like?

– No.

Oh well, Do you know what a warning device is?

– Yes.

Great, A low heat warning device is a kind of warning device.

Its purpose is to provide an output brightness which depends on the input heat intensity.

This component here is a lamp.

Let's go through what the lamp does.

What's the output brightness of the lamp going to be when the input voltage (V7) is high (9V)?

Note that the system uses the difficulty level of the lamp (1) to make the assumption that the user probably knows what it does.

– High (lamp on)

Yes, great, it's high (lamp on).

What's it going to be when the input voltage (V7) is low (0V)?

– None (lamp off)

Yes, splendid, it's none (lamp off).

The system now infers that the user knows about the lamp.

This circuit section here is a inverter.

Do you know what the inverter does for different inputs?

– No.

Note that the difficulty level of 'what the inverter does' is such that the system can't guess whether or not it's known, so asks.

Oh well, let's go through what the inverter does.

When the input voltage (V5) of the inverter is high (9V) the output voltage (V7) will be low (0V).

When it is low (0V) the output voltage (V7) will be high (9V).

– How does the inverter work?

OK, Suppose the input voltage is high (9V).

When the input voltage (V5) of the transistor is high (9V) the output resistance (R4) will be low (fully conducting).

When the potential divider has one average (a few KOhms) input resistance (R3) and one low (fully conducting) resistance (R4) then the output voltage (V7) is low (0V).

Figure 7–11: Explaining a Complex Circuit (cont. on next page)

Suppose the input voltage (V5) is low (0V).

When the input voltage (V5) of the transistor is low (0V) the output resistance (R4) will be high (not conducting).

When the potential divider has one average (a few KOhms) input resistance (R3) and one low (not conducting) resistance (R4) then the output voltage (V7) is high (9V).

Anyway, we were in the middle of describing the components of the low heat warning device.

This circuit section here is a comparator. Let's go through what a comparator does.

When the input voltage (V2) of the comparator is quite high (towards 9V) the output voltage (V5) will be low (0V).

When it is quite low (towards 0V) the output voltage (V5) will be high (9V).

This circuit section is a heat detector unit.

Do you know what the heat detector unit does for different inputs.?

– Yes.

Great, OK, we better explain the behaviour of the low heat warning device.

Suppose the input heat intensity is low (cold).

In the heat detector unit section.. What's the output resistance (R1) of the thermistor going to be when the input heat intensity is low (cold).

Note that the system chooses to go through this in detail because the behaviour of the heat detector is already known.

– quite high.

Precisely, excellent, it's quite high.

What's the output voltage (V2) of the potential divider going to be when there is a quite high input resistance (R1) and a average (a few KOhm) input resistance (R2)?

– quite low.

Correct, splendid, it's quite low.

What's the output voltage (V5) of the comparator going to be when the input voltage (V2) is quite low?

– high (9V).

Yes, very good, it's high (9V).

Dialogue continues

Figure 7–11: Explaining a Complex Circuit (cont.)

“When the input voltage (V5) of the inverter is high (9V) the output voltage (V7) will be low (0V).”

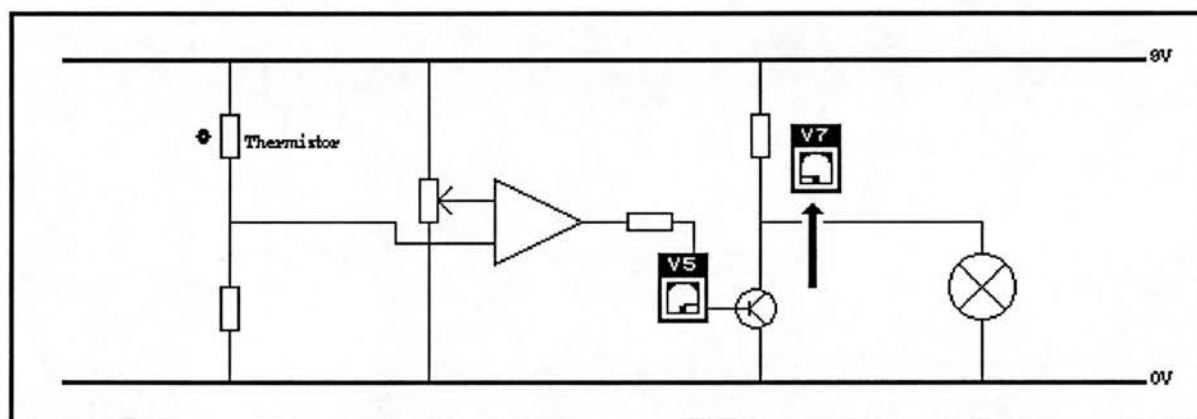


Figure 7–12: Graphical Display in Complex Explanation

be represented in a specialisation hierarchy, where tutoring discourse might be a major type and different styles of tutorial discourse sub-types¹³.

In very different domains more work is needed. Plans to explain how to get somewhere, how to troubleshoot an electrical system, or how to do some task will all look very different. Explaining these requires an understanding of how people explain and of how people learn in these domains. For some types (such as tasks) the results are probably fairly simple — Grosz claims that task dialogues follow the structure of the task [Grosz 77] and though it seems likely that it is more complex than this and that there should be another level of planning above the task representation, a simple task description (such as used in chapter 4) may be adequate. Other domains on the other hand may require substantial empirical analysis.

However, circuit behaviour is by no means a trivial domain. The chapter has shown that the approach to explanatory discourse developed in this thesis does indeed seem to work in a domain of realistic complexity.

7.4 Conclusion

This chapter has shown in detail how the EDGE discourse planner may be used to generate explanations in a particular domain — explaining how simple electronic systems work. This involved developing explanation plans based on analysis of human explanations in that domain and on related research on explaining physical systems. A simple representation for device models was also discussed, though this has many limitations. A number of detailed examples were given, showing how each aspect of the system — content plans, discourse plans

¹³An alternative would be to emphasise how the roles of the participants influence the discourse — these may change within a single discourse genre.

and the user modelling component — combine to generate complex explanatory dialogues.

The particular plans and models developed in this domain satisfy the criteria introduced in section 7.2:

Structure and Types of Explanation: Explanations involve many different types of description (not just causal sequences) and these are partially ordered using prerequisite relationships between concepts.

Value Display and Representation: Qualitative values are used, and these may be displayed on meters in the circuit diagram.

Levels of Explanation: Explanations may be given at different levels, with the user able to ask questions to probe deeper levels.

The main contribution of the work to the problem of generating explanations of how things work is in the recognition that simple qualitative causal explanations are inadequate, and that many different types of explanation must be combined (in some sort of planning process) in order to generate understandable and effective explanations. The work integrates text planning approaches (such as [Paris 87]) with work on generating graphical-textual qualitative causal explanations (e.g., [Forbus & Stevens 81]). The underlying device models and causal reasoning processes are very simple, yet it should be straightforward to incorporate more sophisticated reasoning systems into a discourse planning framework.

Chapter 8

Evaluation, Assessment and Further Work

The main part of this thesis has concentrated on developing a computational model of explanatory discourse based on analyses of human verbal communication. In the last chapter a particular implementation was discussed — tutorial explanatory dialogues about circuit behaviour. This chapter will assess both the theoretical architecture and the particular implementation with a view to further work.

The first section will describe a limited empirical evaluation of the particular implementation described in the last chapter. This will be followed by a discussion of some of the theoretical limitations of the architecture, and a summary of directions for further work.

8.1 Empirical Evaluation

At this point in the thesis it is unclear whether the model developed is a practical approach to the computer generation of complex explanations. The model captures various features of human explanatory dialogues — but is this an acceptable style of presentation in human-computer interaction?¹ This question can only be answered by empirical evaluation.

Of course, there is a limit to what a limited empirical evaluation can hope to achieve. It would be very hard to demonstrate that the system was more effective (pedagogically) than another approach, or that the architecture is necessarily better than another. However, it is possible to obtain some subjective views on a number of aspects of the system and discover how the system is used by different people. From these observations it is possible to assess the potential of the approach and see how the system may be improved.

Aspects of the system which can be evaluated include:

- The particular content plans used.
- The discourse plans used (such as responses to interruptions).
- Remediation plans (responses to errors).
- User model inference rules.

Although none of these particular plans and rules are essential to the basic EDGE model¹, they do represent an important contribution of the research. If practical systems are to be developed these aspects are as important as the overall architecture.

¹This is meant to be a framework for implementing a variety of types of explanations in a variety of domains.

8.1.1 Type of Evaluation

The evaluation consisted of a limited formative evaluation based on a small number of subjects using the system and completing a simple questionnaire.

There were two stages to the evaluation. The first stage served mainly to remove the remaining bugs in the program and improve the interface sufficiently for more unsupervised exploration of the system. This stage involved three subjects using the system for up to half an hour under supervision, then making informal comments loosely based on an initial version of the questionnaire. Between each session the system was debugged and the interface improved where required.

The second stage of the evaluation involved five people using the system in a more unsupervised manner and completing a revised questionnaire. These sessions were scripted. This stage revealed remaining problems, different styles of interaction with the system, and the subjects views on a number of features of the interaction. Details of the subjects and task are given below.

Subjects

The EDGE electronics explainer is very much a prototype system, exploring the ideas in the EDGE model, rather than aiming to develop an effective pedagogical tool. As a result there has been relatively little attention paid to details of the interface, and subjects without a computing background might find it difficult to use. Because of this constraint, all the subjects chosen were members of the department of artificial intelligence. Their knowledge of the domain varied from complete novice to expert, though they typically had learnt some electronics in the past, but did not remember it well.

Several of the subjects had expertise related to the EDGE system, such as knowledge of discourse, language use, explanation or circuit models. Their com-

ments therefore tend to reflect both these areas of expertise and their personal experience of using the system.

Task

The task set the subjects was fairly open-ended. In both stages of the evaluation, subjects were told that there were four circuits that they could get explained, and how to start these explanations. They were all taken through one explanation to illustrate the basic mechanisms and facilities of the system. Then they were allowed to obtain any explanations they wanted until they felt they were familiar with the system and with the circuits. This typically took about half an hour. At the end of the session subjects filled out the questionnaire unsupervised, and then went through their comments with the experimenter, adding more comments and criticisms.

8.1.2 Identifying Problems

Problems with the system could be identified from four sources:

- From comments (written and verbal) made by users during and after the interaction.
- From answers to directed questions in the questionnaire (appendix F).
- From examining the scripts of the sessions, and unstructured observation during the evaluation sessions.
- From examining the user model at the end of the session. This could be compared with the observer's and the user's view of the user's knowledge.

Problems were identified in both the preliminary and in the main stage of the evaluation. They can be divided into four main categories — problems with the

interface, problems with the explanation content, problems with the interactions and problems with the user model. These are summarised below. Where a change was made to the system (following problem identification) this took place in the first stage of the evaluation.

8.1.2.1 Problems with the Interface

In the first stage of evaluation there were a number of minor problems with the interface concerning positioning of menus and inaccuracies in the English. These were improved for the next stage of the evaluation. However, there are many other improvements which should be made before a more realistic evaluation of the system.

Several of the comments concerned the graphics. It was easy to fail to notice graphical actions, and hard to switch focus between the graphics and text windows. These need to be highlighted in some way to attract attention to the relevant physical or linguistic action. A more significant point concerned the co-ordination of text and graphics — this is done at the sentence level (a graphical action can coincide with any sentence being printed), yet in places some users would have preferred to see graphical actions taking place after the associated noun phrase in the sentence. To allow this would require a relatively major change in the system.

8.1.2.2 Problems with the Content

In both stages of the evaluation there were a number of comments about the content of the explanations. There were certain types of explanation missing (such as a description of a component's function in the particular circuit) and explanations failed to make the distinction between a particular component and the general type of component. It was also impossible to ask about the function of the different wires, or about the meaning of voltage or resistance, for example.

As a result, the system is inappropriate for the complete novice as too much background knowledge is necessarily assumed. However, this thesis does not focus on providing adequate models for a wide range of explanation types, and the problems encountered did not get in the way of useful exploration of the system.

One problem with the content that was corrected was the fact that the system would explain things in terms of 'potential divider' circuits even if the user said they didn't know what a potential divider circuit was! This was partially corrected so that the system could explain potential divider circuits near the beginning of the explanation. However, the problem also revealed a gap in the user model indirect inference rules. The system should be able to infer that if a prerequisite fact (what some device 'is') is unknown, then the fact (what some device does) is also likely to be unknown.

Another problem (linking content with interaction) was that the system did not always interpret questions in the same way as the users. This was partly due to badly phrased question menus, and so is an interface issue. However, it also suggests that the users should have the opportunity to say "No, I don't mean that, I mean" and renegotiate the question topic — this sort of correction was occasionally observed in the transcripts. Finally, it may be that the discourse context should influence responses to questions, so a question which means one thing in one context should mean another thing in another context.

8.1.2.3 Problems with the Interaction

Overall people were happy with the style of interaction, but some felt they had insufficient control. In the evaluated version of the system the user cannot repeat sections of the explanation or abandon part of an explanation. This feature has since been added, illustrating the use of an explicit plan-based model of discourse in allowing reference to discourse segments.

There were also two problems with question asking which were corrected. The users could not ask a question as a response to a question, and could not ask a number of questions at the end of the session. The first problem was corrected so that question menus have the option to delay the question. The question is repeated after the interruption. The second problem was corrected by adding a final boundary exchange to the stack if a question is asked at the end of a session (see example interaction in appendix G.1). Both these corrections move away from the strictly hierarchical model of interactions, towards one emphasising local interactions.

One remaining discourse bug was the following. If users asked a number of simple questions in a row these would be treated as nested interruptions (if they had not 'come out' of the interruption yet). This resulted in the following sequence:

— **What's this component?**

This component here is a light dependent resistor.

— **What's this component?**

This component here is a transistor.

— **What's this component?**

This component here is an operational amplifier.

Anyway, we were in the middle of saying what a heat detector circuit does.

Anyway, we were in the middle of saying what it does.

Anyway, we were in the middle of saying what it does.

There were a number of more subjective comments about the interaction. One user (a novice) would have preferred more meta-comments to motivate going through parts of the circuit for the most complex example. The focus shifts too suddenly from the function of the circuit to the behaviour of some arbitrary section. In another case, after an interruption it was not always clear to a user why a question was suddenly asked. It seems likely that the use of meta-comments to direct the focus of attention should depend on the level of understanding (or confusion) of the user.

8.1.2.4 Problems with the User Model

There were a number of problems with the user model inferences — these were in general too conservative, as the system always decreased its assessment of the user's level of expertise as the session progressed. This would tend to happen more with people who asked a lot of questions of the system — each time a question was asked the system adjusts the level of expertise according to the difficulty of the question. As a consequence, an 'expert' who used the system by asking a large number of questions, then quitting the explanation before the system began questioning him, was downgraded to a novice by the end of the session, while a novice who correctly answered a number of questions by guesswork and by examining the past explanation (but who did not ask many questions) did somewhat better.

Another apparent error with the user modelling system arose with consecutive questions. Currently the system assumes the user doesn't know something if they ask a question², but that they 'maybe' know it if it is answered and they acknowledge that answer — yet if they ask another question instead of acknowledging the answer the system is left thinking they don't know the concept. This was a deliberate feature of the modelling system, yet needs to be improved so that later acknowledgements may serve multiple functions — acknowledging that the last utterance is understood, and that therefore the utterance before the interruption is.

So, from the comments and observations of users and from an examination of the user model, several problems were identified in each aspect of the system. Most of these could be fixed by minor modifications of the system, but a few (such as those concerning the explanation content) would require significant extensions.

²This is a somewhat unreliable inference as users sometimes ask questions to 'see what will happen' in a novel system, or just to remind themselves.

8.1.3 Examining the Scripts

Examining the scripts of the dialogues revealed further cases where the system was behaving in a less than optimal manner. One major problem concerns how to revise an explanation plan following an interruption — the current rule of ‘pruning’ goals which correspond to the goals of any interrupting questions works reasonably in many cases, but is not a full answer. Appendix G includes a number of extracts from the scripts along with comments on their features and problems.

Observation and later examination of the scripts also revealed a range of styles of interaction. Some users tended to ask far more questions and to play a dominant role in controlling the interaction, while others would follow through the system’s explanation, just asking a few questions at the end. A mixed initiative system such as the EDGE model can support these different styles of interaction to some degree, but begins to fail when the user wants to take more control. The user modelling inferences become less certain when the system does not have the chance to ask many questions, and it is harder for the system to maintain a coherent plan of what to explain. The evaluated system also did not allow the user full control of the explanation (such as being able to quit some sub-topic).

8.1.4 The User’s Opinion

The questionnaire given to the users at the end of the session gives an indication of their subjective view of the system. The results (from the five questionnaires) are summarised below:

- All subjects found the explanation to be coherent, and all but the two novices considered it to be at the right level of difficulty (the novices found the most complex circuit to be too hard). However, this reflects more on the domain and the particular users than on the system’s ability to tailor

the explanation. Several users wanted to ask things which the system could not handle. These included questions about the meaning of terms, the functions of the wires, and the function of components in achieving a devices overall purpose.

- Most users found the dialogue natural (apart from the ‘anyway anyway’ bug). Two users wanted more control — such as being able to repeat or quit sub-topics.
- Most users found the discourse markers and meta-comments helpful and the English (including pronoun use) acceptable³. However, several mentioned minor problems with the graphics.
- Almost all users thought that asking questions, being asked questions and having coordinated text and graphics were all important in an explanation of this sort. A novice (not familiar with circuit diagrams) rated graphics unimportant, and the relative expert who had a ‘question-asking’ style of interaction rated being asked questions unimportant.
- All users enjoyed using the system (to varying degrees).

This doesn’t tell us very much — the more useful information comes from examining the scripts and in the particular problems identified by the users. However, it is enough to suggest that the system is acceptable to users and that it has potential as a practical approach to generating explanatory dialogues.

³The exceptions were mainly in my use of pronouns/markers/apostrophes in fixed templates!

8.1.5 Conclusion

The evaluation discussed here is clearly very much a preliminary exploration of the potential and the problems of the system. From it we can conclude:

- It has potential: people enjoyed using the system, had no major problems with the interaction, and found many of its features helpful.
- It has problems: The scripts and answers to the questionnaire highlighted a number of problems with the system. Many of these concerned the interface and most of the others require minor changes consistent with the general architecture. However, there are some general problems, such as how to give the user more sense of control in a mixed initiative interaction, and how to revise and modify plans based on user initiatives. It is at least possible to explore these issues within the EDGE framework, but there are no simple answers.

This evaluation has not considered whether the system is an effective pedagogical tool — this would only become appropriate if the EDGE architecture was used in developing a complete explanation system taking more fully into account pedagogical goals and strategies.

8.2 Assessment and Comparison

The empirical evaluation discussed above gives an indication of the problems and limitations of the particular implementation. However, it is also important to consider the inherent limitations in the architecture⁴, and how these can be remedied.

⁴By architecture we mean here the actual planner and plan formalism and the general approach to user modelling.

There are obviously a number of limitations which are not addressed by the theory, and so are not relevant to the discussion. These include problems of natural language understanding, using multiple models and answering a wide range of question types. This section will therefore concentrate on restrictions of the basic EDGE model — how it plans discourse, controls interactions and updates a model of the user's knowledge. Some of these restrictions turn out to be unimportant. However, there are several serious limitations which must be addressed.

8.2.1 Limitations in the Planning Formalism

The planning formalism used is extremely simple. Skeletal plans⁵ are used to incrementally plan the discourse. There are two major restrictions inherent in this approach. First, skeletal planning is less powerful than some other approaches, especially where there are no prebuilt partial solutions to a problem. Second, incremental planning does not guarantee an optimal or even a complete solution where there is one. It is impossible to use later planning decisions to influence earlier ones. The rest of this section will consider these two limitations further and discuss how more flexible planners could be developed.

8.2.1.1 Skeletal Planning

The use of skeletal plans is similar to Murray's curriculum planner [Murray 89] and contrasts with Peachey and McCalla's [Peachey & McCalla 86] which is

⁵Simple skeletal plans are extended to allow preconditions and constraints. However, because each plan has a single effect, goals and effects are equated. Satisfying preconditions is therefore equivalent to a conditional call to a sub-plan, and so the formalism (though superficially like more powerful planning systems) is in fact a simple extension of skeletal planning where it is determined in advance how to teach each concept in the domain.

based more on a conventional STRIPS type planner [Fikes & Nilsson 71]. These two types of system have very different capabilities. For example, a simple skeletal planner cannot reason about operators with multiple effects. However, the operators used by Peachey and McCalla all have a single effect — for example, the operator ‘TEACH-COD’ has the effect ‘add SK(COD)’ (add the fact that the student knows fact COD to the student model). All of these could be represented in the EDGE skeletal plan formalism, using only the preconditions part of the plans. So, though there are restrictions compared with conventional planners, in general this does not seem to be important — it is more natural to reason in terms of a set of teaching operators to teach different concepts than in terms of transforming a representation of the student’s knowledge.

The basic skeletal planning approach has been widely used. Moore, for example bases her text planner on skeletal plans for different rhetorical relations [Moore & Swartout 88b]. In most such systems, planning proceeds by a mixture of goal expansion (into detailed plans) and goal refinement into more specialised versions of plans applicable in different situations. Although it does have some theoretical limitations, these do not seem important, at least in well specified, well understood domains.

8.2.1.2 Incremental Planning

Both Moore’s system and Murray’s build a complete plan before executing⁶. This has the advantage that it is possible to obtain the best plan (assuming that the state of the world remains constant). Planning incrementally has the inherent problem that one cannot predict problems that will arise in later planning and use that knowledge in making choices. Incremental planning though, is efficient and takes into account the unpredictably changing situation. When planning

⁶Murray does in fact suggest that incremental planning would be more appropriate.

incrementally can be guaranteed to find a complete plan, satisfying all the given constraints, then this seems an appropriate approach. However, when this is not the case it may be necessary to search the space of partial plans before the execution begins.

Although incremental planning appears adequate in the current domain, it would be worth exploring how the amount of planning ahead can be varied depending on the factors discussed above. This would be a step towards providing a planner which was both robust and efficient.

8.2.1.3 Increasing Flexibility

The EDGE planner interprets subgoals in a fixed order and uses a fixed decision to decide on whether to expand preconditions. However, if true adaptive dialogues are to be achieved then it should be made more flexible. Moore's planner, for example, is less careful to choose plans which definitely satisfy all the constraints on the user model. If a precondition is not definitely known, then a plan may still be selected, on the basis that the user may always ask for a clarification. Elsom-Cook, on the other hand, uses a large set of heuristics for selecting the next topic given the state of the discourse, the user model and curriculum (sub-topic) relations between topics [Elsom-Cook 85]. These include a numeric weighting on the importance or relevance of the topics.

One obvious extension to the EDGE planner which would increase its flexibility would be to view subgoals of plans as unordered unless explicitly stated. The goal stack will then become more like an agenda with a set of goals to be satisfied and a number of ordering relations between them. Which goal is satisfied next may then be influenced by a number of other criteria, such as local coherence and importance. This would be similar to Hovy's recent extension of his 'Rhetorical Structure Theory' based planner, which allows focus constraints to influence partially ordered *growth points* in the plans [Hovy & McCoy 89],

and Jullien and Marty's work on dialogue plan-revision following *deviations* in advice dialogues [Jullien & Marty 89].

Another extension would be allow it to vary the care with which it describes prerequisite knowledge. For some users and in some discourse situations it may be appropriate to assume that they will ask if they have missing information, while in other situations it may be important that it is immediately understandable.

Finally, the plans themselves should be made more flexible, so that the dialogue style could change depending on the discourse situation.

8.2.1.4 Conclusion

The plan formalism used has a number of limitations. However, it provides a starting point for the principled development of more flexible discourse planners. The most important extension of the basic formalism is probably to allow partially ordered subgoals in plans so other constraints may influence the progress of the discourse.

8.2.2 Local Responses in a Changing Situation

The above section has looked at the limitations of the planning framework compared with other related formalisms. But how far is it appropriate to impose a high level structure on the dialogue in a dynamically changing situation? This section will look at some of the problems with any planning approach and how local responses can be managed within a plan-based framework.

8.2.2.1 Interruptions and Dialogue Roles

In the current model, the system always continues a planned explanation after an interruption. Yet the situation (user model and discourse context) may have

changed radically during the interruption. The change in the situation will influence the detailed planning of the continuing dialogue — but the whole plan may now be inappropriate.

Where one participant (the system) dominates the dialogue with well defined goals, this model of dialogue may be appropriate. However, where there are more equal dialogue roles an approach emphasising local coherence may be more appropriate, where the expert's continuing explanation should follow on from the interruption. One possible move towards this, but retaining the idea of an overall explanation plan, is to use partially ordered plans with local coherence (focus) influencing plan ordering. However, there may be situations where even this degree of planning ahead is inappropriate.

8.2.2.2 Using the Discourse Situation in Plan Selection

The current system also fails to use the discourse context (apart from the user model) to select content plans⁷ or influence English expression — yet the goal which led to something being explained should influence how you explain it. For example, the system may decide to describe a device's behaviour as part of a causal explanation, as part of a componential description or as a general summary within a complex explanation of that device's behaviour. These different contexts should perhaps influence how it was described. Though it would be possible to use the discourse context in this way, selecting different versions of a plan depending on higher level goals, this is not currently done.

Another case where the discourse model should influence plan selection is in determining the appropriate responses to questions. Currently, each ques-

⁷There is one special case where the discourse context is used to influence content selection — if a sub-plan has previously been called as a question the plan is not expanded. This avoids the system re-explaining things the user has asked about, but is over-simple.

tion type has a single explanation goal associated with it — yet the discourse context may influence what people really mean by a question, and what information they want in the response. Moore and Swartout have made some suggestions concerning how the discourse context may influence response planning [Moore & Swartout 88a, Moore & Swartout 88b].

8.2.2.3 Explanation Failure and Repair

Another major gap in the current model is its inability to replan part of an explanation if it fails, and hence respond appropriately to problems in understanding. A fairly straightforward extension would be to allow the student to indicate at any point that they don't understand. Then the system would examine the discourse model to find either preconditions of plans which might not really be satisfied, or constraints that might not be true due to an uncertain user model. Having found the possible problems with the plan, the system might negotiate with the student to identify the exact area, and re-explain whatever is required. This would be similar in many ways to Moore and Swartout's work, but would allow interruptions at any point, not just at the end of an explanation, and would include the negotiation of the problem area. The aim would be to produce dialogues such as the following:

.... so the output voltage is high.
— **Huh?**
Don't you know what a potential divider is?
— **No.**
Oh, OK, I'll go through that.

Sometimes, of course, the source of the difficulty may lie in the immediately preceding utterance (such as the use of some technical term). The system would have to guess at possible problems and check with the user. Negotiation becomes more and more important where there are many potential sources of confusion

and an unreliable user model, and systems which seek to reliably guess the cause of the problem with no interaction are unlikely to succeed.

8.2.2.4 Conclusion

This section has shown how local responses can be managed in a plan-based framework which depend on the emerging situation. While accepting the criticisms of [Suchman 87] for example, concerning the limitations of plan-based systems and the importance of locality and situationality, this thesis argues that these may be realistically combined with a flexible plan-based approach. Interruptions and remediation are already possible — repair, local coherence and further use of the discourse context are other important steps.

8.2.3 Limitations of the User Model

The EDGE user modelling system is essentially based on the overlay paradigm. It assumes that the student's knowledge is a subset of the expert's, and does not represent misconceptions or alternative viewpoints. This is obviously a major restriction, but seemed justified in the context of generating explanatory discourse. Just because misconceptions aren't explicitly represented that does not mean they cannot be reacted to. Remediation strategies map error types to the plans used to attempt to correct them, so there is an implicit knowledge of misconceptions in providing immediate feedback to errors.

There would of course be some advantages in representing misconceptions, and certainly advantages in being able to reason about alternative viewpoints. For example, in electronics both the student's questions and answers to questions might be based on a 'current' based model of electricity (an alternative view), or a model based on circuit equations rather than system behaviour. In a tutoring context it would be important to be able to react appropriately when these alternative models are used, sometimes switching back to the student's model, or

persuading them that another model is more appropriate. Also, if misconceptions are detected it may not always be appropriate to respond immediately, even within a short dialogue. These misconceptions should be represented in the user model.

There are also many other parameters which could be included in a user model. There could be an overall measure of success rate in answering questions or indications of preferred dialogue style. These could be linked with more flexible dialogue control to generate more reactive dialogues.

So, there are a large number of issues which could be addressed about user modelling in the context of the EDGE system, especially when used to generate tutorial discourse. Some of these have been mentioned above, or in chapter 6. These are all active research areas in the artificial intelligence and education community.

8.2.4 Summary

This section has attempted to point out a number of limitations and restrictions in the EDGE system, and begun to point out how these could be removed. Particular areas of further work are summarised in the next section.

8.3 Further Work

This final section will identify a number of concrete areas for further work, based largely on the theoretical and practical limitations highlighted in the preceding discussions. Some of these points may have been introduced above or in previous chapters, but are summarised below.

In much of this discussion, further work can be considered at several levels — at the level of improving the existing system; at the level of extending and generalising it; and at the level of the EDGE architecture.

8.3.1 Explanation Content

8.3.1.1 Improving the System

There are obviously a large number of extensions that could be made to the content of the explanations of circuits, and to the range of questions answerable by the system. These include 'lower level' explanations suitable for novices, explanations which take into account the function of a device in a particular circuit (cf. [deKleer 84, deKleer & Brown 83]) and more general descriptions of device behaviour (such as saying that a comparator compares its two inputs, rather than describing its particular input-output behaviour).

Another major extension would be to consider the affects of the discourse situation which motivated the explanation, such as the underlying goals of the speaker or hearer. An explanation given as part of a consultation dialogue or an advice session must consider the affect of this wider discourse context. This extension would require further analysis of explanations to study the influence of the discourse context, but would probably not require a change in the architecture.

8.3.1.2 Generalising the Plans

The plans for explaining circuit behaviour are very similar to the schemas used by Paris for describing physical systems [Paris 87]. This suggests that the plans could be generalised to a much wider domain. The representation makes this very easy — plans may have constraints which refer to the type of device (given a device-type hierarchy), so may be re-used and generalised as required.

The plans also seem to be instances of more general rules for explanations or descriptions in any domain. For example, the structure of stories often begins with an abstract and orientation and ends with a summary (e.g., [Labov & Fanshel 77]) while the structure of advice in help systems may start

with an announcement and some contextual information and end with an evaluation [Breuker *et al* 87]. The explanations of circuits start with an announcement (from the discourse plans) and some information to link with existing knowledge and end with a conclusion. A general theory of explanation should consider how the schemas and structures in particular domains fits into these more general explanatory patterns.

8.3.1.3 Improving the Architecture

As has been mentioned in section 8.2, not all sub-goals of plans have a fixed order, yet the order is assumed fixed by the system. Allowing partially ordered plans and an interpreter which took account of other factors (such as local coherence) when selecting which goal to expand would result in a more flexible, reactive system.

Brecht, in her tutorial planner uses a wider range of possible links between topics (than simple prerequisite and sub-skill) and uses heuristics to select topics based on these links [Brecht *et al* 89]. It is possible that more flexibility in planning would be possible with this sort of representation.

8.3.2 Interactions and Dialogue

8.3.2.1 Improving the System

There are several minor improvements that could be made to allow more flexibility in the dialogue. It has already been extended to allow the user to quit or repeat sub-topics. A more complex extension would be to allow users to indicate at any point that they don't understand, and have the system reason about likely causes of confusions, negotiate the exact cause of trouble, and re-explain the problem area. This would require recording assumptions about the user's knowledge in the discourse model and using that to guess possible problems.

8.3.2.2 Generalising the Plans

Currently the discourse plans used give a fixed interaction style (tutorial interactive explanations). Yet many of the discourse plans generalise to other discourse genres. Just as the content plans may be generalised so they work for a range of domains, so may the discourse plans be generalised for a range of styles of explanation appropriate in different situations. If the factors influencing the structure of the interactions could be identified, then a more flexible approach to dialogue control could be developed, switching dialogue strategies depending on the emerging situation. This might include different styles of questioning, different use of meta-comments and different responses to questions.

As well as generalising the plans, the planner/interpreter should be made more flexible to allow different types of planning. For example, the current system expands prerequisite goals if it is not fairly certain that they are understood. An alternative strategy would be to only expand them if they were definitely not understood, and leave it up to the user to ask any further questions if this results in missing prerequisite information.

Finally, the plans should be generalised to allow more equal dialogue roles between system and user. It should be possible to support cooperative explanations where each participant may contribute questions, statements and checking moves. Petrie-Brown and Baker both argue for equal dialogue roles in tutorial interaction [Baker 89, Petrie-Brown 89].

8.3.2.3 Improving the Architecture

The current system controls interactions by imposing a hierarchical structure on the discourse. It seems likely that at least some of this structure should emerge out of the situation [Suchman 87] and not be part of a global plan. For example, topic closings (e.g., 'Is that enough about how the circuit works?') might be more appropriate as a local decision to negotiate the close of the topic, rather than as

part of a high level plan. In moving towards more flexible interactional styles, the EDGE model may further abandon a rigid hierarchical discourse structure.

8.3.3 The User Model

8.3.3.1 Improving the System

The evaluation above has shown that there are problems with the user model inference rules used, though the abstraction hierarchy of concepts, and most of the inference rules seem to work well in at least achieving a range of explanations and dialogues depending on the user. Further evaluation will be needed to confirm whether or not the user modelling system is making useful and largely correct inferences, given the minor corrections required⁸.

8.3.3.2 Improving the Architecture

There are a huge number of ways the user modelling system could be developed, such as representing misconceptions, or allowing the different sources of information to contribute to the model with various degrees of certainty.

8.3.4 Applications

The above points have shown how the EDGE system and architecture may be improved in various ways. Yet the EDGE system seems to provide a framework

⁸Of course, in all issues concerning adaptivity to the user — whether the dialogue style, content or surface text — there is the open question regarding the tradeoff between adaptivity and consistency in human-computer interaction, discussed by [Hagglund 81] among others. This thesis assumes that modelling human adaptivity in dialogue is a useful approach, but this is by no means confirmed.

for more applications oriented research. There are several areas which could be developed. It provides a framework for interactive text [Cawsey 89c] and explanation [Cawsey 88b] generation; for structuring tutorial discourse [Cawsey 89d] and for generating help and advice in human-computer interaction [Cawsey 89b]. Each of these areas could be developed further. For example, if it was used for generating explanations and dialogues in a tutorial system, then more attention should be paid to pedagogical issues and to the higher level planning of tutorial interactions. If it was used as a general approach to text generation then it should be integrated further with the currently dominating approach of 'Rhetorical Structure Theory'. If it was used in a help and advice system then there would be the issue of how to detect problems and less than optimal performance in a non-intrusive manner and use this to influence the explanation content.

Any of these would be fruitful directions for further research. It would involve integrating the EDGE model with current research ideas in the particular areas and interleaving further evaluation with development. The work described in this thesis has deliberately avoided concentrating too far on a particular application, or using a particular established approach. Instead it aims to develop a more general model by the integration of ideas from different areas. The evaluation described in this chapter provides a further source of insights, rather than being an assessment of an application.

8.4 Conclusion

This chapter has discussed a number of problems and limitations with the EDGE architecture and with the EDGE tutorial electronics explanation system. It has concluded with a summary of areas of further work which may remedy some of these limitations and generalise the architecture.

Some of these suggestions for further work are relatively straightforward but important extensions to the system. These include allowing partially ordered

plans; implementing *repair* strategies; generalising the content and discourse plans for more general domains and types of discourse; and implementing more flexible dialogue strategies using these generalised plans and a flexible planner. These are seen as the most important and immediate extensions to the EDGE model.

Chapter 9

Conclusion

This final chapter will summarise what has been presented in the previous chapters, and conclude with an assessment of the contribution of the work and a particular direction to pursue in further work.

9.1 Summary — The Story So Far

This thesis started with the problem of generating interactive explanations which depended on the user's knowledge (chapter 1). No existing approach seemed to offer a complete solution, but there were a number of relevant and important areas (chapter 2). An analysis of human explanations of circuit behaviour showed that the content of such explanations could be viewed as planned, based on prerequisite and sub-skill relations between the concepts being explained (chapter 3). Furthermore, it showed that any dialogue with the user could be seen as based on this planned content, with interruptions and follow-up questions from the novice. It might involve the expert questioning and testing the novice, but the subject matter of these questions would be part of the planned content. The

structure of the content being explained appeared largely independent of the structure of the interactions¹.

In chapters 4-6 the EDGE model of explanatory discourse was developed. This started with a discussion of how to generate communicative text (chapter 4). The EDGE approach to text generation emphasises how text can be planned so that it can be understood by a particular hearer, rather than using schemas or rhetorical relations to generate coherent text. Two levels of text planning were presented — content plans are used to decide what to say, while discourse plans are used to decide how to present this, such as deciding on the use of discourse markers and meta-comments. The discourse planning level was developed in chapter 5, which discussed how to control the dialogue in interactive 'text' generation. The discourse planner now takes responsibility for structuring the interactions and deciding on dialogue moves. Chapter 6 concluded this discussion of the EDGE discourse generator by showing how a model of the user's knowledge could be updated as the explanatory dialogue proceeded. The user model (and the goal of updating it) influences both the content of the explanation and the dialogue with the user.

The EDGE model developed in chapters 4 to 6 is a general approach to generating explanatory discourse, not specific to a particular domain or discourse type. Chapter 7 showed how the EDGE model is used for a particular domain (how circuits work). The content plans are based on the analysis in chapter 3, and on related research in the domain. Several examples were given of explanations generated by the system.

The EDGE model is fairly simple — there are still many theoretical and practical limitations. Some of these were described in chapter 8, which included a description of a small scale evaluation of the system. This evaluation suggested

¹These observations are based on a fairly limited and subjective analysis of the explanations, so only suggest what the EDGE system should be modelling.

that the EDGE system has potential as a practical approach to generating explanatory discourse, but highlighted some of the problems which first must be overcome. Several detailed suggestions for further work were made which would allow the user more control over the explanation while retaining a global plan of what should be explained.

So, the EDGE model has come from an analysis of human explanatory dialogues, to a partially evaluated flexible architecture for generating such dialogues. The model uses knowledge of discourse structure and plans of how to explain in a particular domain to generate flexible, mixed-initiative interactive explanations. But what does this contribute to our understanding of explanatory discourse, or to the building of such systems?

9.2 Where Has it Got Us?

The contributions of the EDGE model can be examined at several levels — the system, the approach and the key ideas.

9.2.1 The System: Informative, Interactive, Individualised Explanations

The EDGE model satisfies the three key criteria presented in chapter 1, and developed in chapters 4-6. Explanations may be generated which are:

Informative — based on prerequisite relations representing how people learn in a particular domain.

Interactive — involving many types of dialogue move.

Individualised — based on a continually updated model of the user's assumed knowledge.

The system appears to have potential as a practical approach to generating explanatory dialogues of different types in different domains.

9.2.2 The Approach: Integrating Ideas

The EDGE model integrates ideas and approaches from different areas. The approach to content planning is very similar to some recent work on curriculum planning in Intelligent Tutoring System [Murray 89, Lesgold 88], yet it is being applied to generating relatively small sections of text — the unit of knowledge transfer being the exchange, rather than the lesson. The idea of using two levels of planning (discourse and domain) has been used in natural language understanding [Litman 85] and is implicitly used in some approaches to tutorial dialogues (e.g., [Woolf & Murray 87])². However, the EDGE model makes these two levels of planning explicit, allowing a great deal of flexibility in discourse style and content. The approach to user modelling is not very original in itself (though it allows descriptions of users knowledge at different levels of detail), yet few previous systems have shown how the user model influences the dialogue and content and how it is updated in the process — it is insufficient to consider the processes of inferring and using a user model separately as they are integrally related. Finally, unlike many tutorial systems, the EDGE model is consistent with a current linguistic model of discourse structure [Grosz & Sidner 85] — this allows a better theoretical treatment of pronouns, discourse markers, interruptions and meta-comments.

²Brecht also suggests using discourse and content plans to plan tutorial discourse, but has not yet implemented such a system [Brecht *et al* 89].

9.2.3 The Key Ideas

So, what are the main points that can be taken from the work? The approach is multidisciplinary, and so contributes to different fields. The following points suggest the key ideas in the different areas:

Tutoring Systems:

— Explanation is a microcosm of the whole tutorial process and cannot be relegated to an ‘interface’ issue. It involves knowledge of tutorial discourse structure, user modelling and ‘remediation’ strategies as well as how to present the underlying knowledge.

Text Generation:

— Text generation should be based on human verbal communication as much as written. It should be treated as an interactive process, taking into account the prior knowledge of the hearer and updating this as it proceeds.

Explanation and Help:

— Explanation is an interactive process. The EDGE model provides a concrete model which could be used in generating interactive explanations which depend on the user’s knowledge. It is not sufficient to consider just the interaction (e.g., [Frohlich 88]) or just the knowledge of the user (e.g., [Wallis & Shortliffe 84]).

Discourse and Text Structure:

— Both domain and discourse plans are important in modelling discourse, whether monologue or interactive. In explanatory discourse the structure of the domain knowledge presented is related to the topic structure in the domain. The discourse plans are consistent both with models of interaction (e.g.,

[Sinclair & Coulthard 75]) and with text structure ([Grosz & Sidner 85]), integrating these ideas via the topic structure in the domain.

9.3 Where Now?

One of the key problems in generating mixed initiative planned discourse is how the interactions with the user should influence the subsequent discourse. For example, how should an explanation be resumed after an interruption? In the EDGE system, the explanation will always resume where it left off, though the changed state of the user model may influence the details of the subsequent explanation, and the use of meta-comments and repetition may make the resumption easier to follow. The model only begins to address the general problem, which may be summarised as:

“Given a plan of what needs to be explained, how should the interactions with the user (questions asked and answered) influence the subsequent discourse?”

This question subsumes problems of explanation repair, local coherence, user modelling and more flexible dialogue strategies. Answering this question will contribute to a model of explanatory discourse which is more reactive to the user, responding to their confusions, questions, misconceptions and suggestions in a constructive manner. Chapter 8 included a number of concrete suggestions in this direction which are consistent with the EDGE model, requiring relatively straightforward extensions. However, developing these ideas will inevitably reveal new problems and provide new insights.

Bibliography

- [Aikins 83] J.S. Aikins. Prototypical knowledge for expert systems. *Artificial Intelligence*, 20:163–210, 1983.
- [Allen & Perrault 83] J.F. Allen and C.F. Perrault. Analysing intention in utterances. In B. Grosz, K. Spark Jones, and B.L. Webber, editors, *Readings in Natural Language Processing*, pages 441–458, Morgan Kaufmann, 1983.
- [Appelt 82] D.E. Appelt. *Planning Natural Language Utterances to Satisfy Multiple Goals*. Technical Report 259, SRI International, 1982.
- [Baker 89] Mike Baker. A model for tutorial dialogues based on critical argument. In Dick Bierman, Joost Breuker, and Jacobijn Sandberg, editors, *Artificial Intelligence and Education*, pages 2–8, 1989.
- [Berry 75] Margaret Berry. *An Introduction to Systemic Linguistics: 1. Structures and Systems*. B.T. Batsford Ltd, 1975.
- [Black & Wilensky 79] J.B. Black and R. Wilensky. An evaluation of story grammars. *Cognitive Science*, 3:213–230, 1979.
- [Bobrow et al 77] D.G. Bobrow, R.M. Kaplan, M. Kay, D.A. Norman, H. Thomson, and T. Winograd. GUS: a frame-driven dialogue system. *Artificial Intelligence*, 8:155–174, 1977.
- [Brecht et al 89] Barbara Brecht, Gordon McCalla, Jim Greer, and Marlene Jones. Planning the content of instruction. In Dick Bierman, Joost Breuker,

- and Jacobijn Sandberg, editors, *Artificial Intelligence and Education*, pages 32–41, 1989.
- [Breuker *et al* 87] J. Breuker, R. Winkels, and J. Sandberg. A shell for intelligent help systems. In *Proceedings of the 10th International Conference on Artificial Intelligence*, pages 167–173, 1987.
- [Brown *et al* 82] J.S. Brown, R.R. Burton, and J. de Kleer. Pedagogical, natural language and knowledge engineering techniques in SOPHIE I, II and III. In D. Sleeman and J.S. Brown, editors, *Intelligent Tutoring Systems*, New York: Academic Press, 1982.
- [Burton & Brown 82] R.R. Burton and J.S. Brown. An investigation of computer coaching for informal learning activities. In D. Sleeman and J.S. Brown, editors, *Intelligent Tutoring Systems*, New York : Academic, 1982.
- [Burton 81] D. Burton. Analysing spoken discourse. In M. Coulthard M. Montgomery, editor, *Studies in Discourse Analysis*, RKP, 1981.
- [Carberry 83] S. Carberry. Tracking user goals in an information seeking environment. *Proceedings of the National Conference on Artificial Intelligence*, 59–63, 1983.
- [Carbonell 70] J.R. Carbonell. AI in CAI: An artificial intelligence approach to computer assisted instruction. In *IEEE Transactions on Man Machine Systems*, 1970.
- [Carbonell 82] J.G. Carbonell. *Meta-Language Utterances in Purposive Discourse*. Technical Report CMU-CS-82-125, Carnegie Mellon University, 1982.
- [Card *et al* 83] S.K. Card, T.P. Moran, and A. Newell. *The psychology of human computer interaction*. Lawrence Erlbaum, 1983.
- [Carlson 83] Lauri Carlson. *Dialogue Games: An Approach to Discourse Analysis*. D. Reidel Publishing, 1983.

- [Cawsey 88a] Alison Cawsey. Explaining the behaviour of simple electronic circuits. In *Proceedings of the International Conference on Intelligent Tutoring Systems*, pages 372–378, 1988.
- [Cawsey 88b] Alison Cawsey. Generating tutorial explanatory dialogues. In *Proceedings of the fourth Alvey Explanation Workshop*, pages 87–101, 1988.
- [Cawsey 89a] Alison Cawsey. A computational model of explanatory discourse: local interactions in a plan-based explanation. In *Proceedings of the Computers and Conversation Colloquium*, Sept 1989.
- [Cawsey 89b] Alison Cawsey. Explanatory dialogues. *Interacting with Computers*, 1:69–72, 1989.
- [Cawsey 89c] Alison Cawsey. Generating explanatory discourse. In *Proceedings of the Second European Workshop on Natural Language Generation*, 1989.
- [Cawsey 89d] Alison Cawsey. The structure of tutorial discourse. In Dick Bierman, Joost Breuker, and Jacobijn Sandberg, editors, *Proceedings of the fourth International Conference on Artificial Intelligence and Education*, pages 47–53, 1989.
- [Chin 88] D. Chin. *Intelligent Agents as a Basis for Natural Language Interaction*. Technical Report UCB/CSD; 88/1396, University of California, Computer Science Div., 1988.
- [Clancey 83a] W. Clancey. *Knowledge-based tutoring: the GUIDON system*. MIT press, 1983.
- [Clancey 83b] W.J. Clancey. The epistemology of a rule based system - a framework for explanation. *Artificial Intelligence*, 20:215–251, 1983.
- [Clancey 88] W.J. Clancey. The role of qualitative models in instruction. In J. Self, editor, *Artificial Intelligence and Human Learning*, pages 49–68, London Chapman and Hall, 1988.
- [Clark & Clark 77] H.H. Clark and E.V. Clark. *Psychology and language : an introduction to psycholinguistics*. 1977.

- [Collins & Gentner 83] Collins and Gentner. Multiple models of the evaporation process. *Proceedings of the 5th Annual Conference of the Cognitive Science Society*, 1983.
- [Collins & Stevens 82] A. Collins and A.L. Stevens. Goal strategies of inquiry teachers,. In R.Glaser, editor, *Advances in Instructional Psychology*, Lawrence Erlbaum, Hillsdale, 1982.
- [Collins et al 86] Allan Collins, John Seely Brown, and S.E. Newman. *Cognitive Apprenticeship: Teaching the Craft of Reading, Writing and Mathematics*. Technical Report 6459, Bolt Beranek and Newman, 1986.
- [Coulthard & Montgomery 81] M. Coulthard and M. Montgomery, editors. *Studies in Discourse Analysis*. London: Routledge and Kegan Paul, 1981.
- [Cox et al 88] B. Cox, J. Jenkins, and E. Pollitzer. An explanation-driven, understanding-directed user model for intelligent tutoring systems. *IEEE Colloquium on Intelligent Tutoring Systems*, 1988.
- [Cullingford et al 82] R.E. Cullingford, M.W. Krueger, M. Selfridge, and M.A. Bienkowski. Automated explanations as a component of a computer aided design system. *IEEE Transactions on Man Machine and Cybernetics*, 1982.
- [Dale 86] Robert Dale. *The Pronominalisation Decision in Natural Language Generation*. research paper 276, Department of Artificial Intelligence, University of Edinburgh, 1986.
- [Dale 89] Robert Dale. Generating recipes : an overview of epicure. In *Proceedings of the Second European Workshop on Natural Language Generation*, 1989.
- [Davey 74] A. Davey. *The formalisation of discourse production*. Unpublished PhD thesis, Edinburgh, 1974.
- [deBeaugrande & Dressler 81] R. de Beaugrande and W. Dressler. *Introduction to Text Linguistics*. Longman, 1981.

- [deKleer & Brown 83] Johann de Kleer and J. Seely Brown. Assumptions and ambiguities in mechanistic mental models. In D. Gentner and A. Stevens, editors, *Mental Models*, pages 155–190, Hillsdale, N.J.L. Erlbaum, 1983.
- [deKleer 84] J. de Kleer. How circuits work. *Artificial Intelligence*, 24:205–280, 1984.
- [deKleer 86] J. de Kleer. An assumption based truth maintenance system. *Artificial Intelligence*, 28:127–190, 1986.
- [Derry et al 88] S. Derry, L. Hawkes, and U. Ziegler. A plan based opportunistic architecture for intelligent tutoring. In *Proceedings of ITS-88*, pages 116–123, 1988.
- [Deutsch 74] B. Deutsch. *Transcripts of Task Oriented Dialogues*. Technical Report 146, SRI International, 1974.
- [Draper 87] S.W. Draper. A user-centred concept of explanation. In *Proceedings of the Second Alvey Explanation Workshop*, 1987.
- [Edmondson 81] W. Edmondson. *Spoken Discourse : A Model for Analysis*. Longman, 1981.
- [Elsom-Cook 85] M. Elsom-Cook. Towards a framework for human-computer discourse. In P. Johnson and S. Cook, editors, *People and Computers: Designing the Interface*, pages 380–393, Cambridge, 1985.
- [Falkenhainer & Forbus 88] B. Falkenhainer and K.D. Forbus. Setting up large-scale qualitative models. In *Proceedings of the 6th National Conference on Artificial Intelligence*, 1988.
- [Fawcett et al 88] R.B. Fawcett, Anita van der Mije, and Carla van Wissen. Towards a systemic flowchart model for discourse structure. In R.P. Fawcett, editor, *New Directions in Systemic Linguistics : Theory and Application*, pages 116–143, Francis Pinter, 1988.

- [Ferrari & Reilly 86] G. Ferrari and R. Reilly. A two level dialogue representation. *Proceedings of COLING '86*, 42-45, 1986.
- [Fikes & Nilsson 71] R.E. Fikes and N.J. Nilsson. STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 2:189-208, 1971.
- [Finin & Drager 86] T. Finin and D. Drager. Gums: a general user modelling system. In *Proceedings of the sixth Canadian Conference on Artificial Intelligence*, pages 24-30, 1986.
- [Forbus & Stevens 81] K. Forbus and A. Stevens. *Using Qualitative Simulations to Generate Explanations*. Technical Report 4490, Bolt Beranek and Newman, 1981.
- [Frederiksen & White 88] J.R. Frederiksen and B.Y. White. Intelligent learning environments for science education. In *Proceedings of ITS-88*, pages 250-257, 1988.
- [Friedland & Iwasaki 85] P.E. Friedland and Y. Iwasaki. The concept and implementation of skeletal plans. *Journal of Automated Reasoning*, 1:161-208, 1985.
- [Frohlich & Luff 89] David M. Frohlich and Paul Luff. Conversational resources for situated action. In *Proceedings of the CHI '89 Conference*, 1989.
- [Frohlich 88] David Frohlich. Conversational dynamics for emergent explanations. In *Proceedings of the fourth Alvey Explanation Workshop*, pages 112-123, 1988.
- [Gagne & Briggs 78] R.M. Gagne and L.J. Briggs. *Principles of Instructional Design*. Holt, Rinehart and Winston, 1978.
- [Gentner & Stevens 83] A. Gentner and A. Stevens, editors. *Mental Models*. Hillsdale, N.J.L. Erlbaum, 1983.
- [Gilbert 87a] G. Nigel Gilbert. Advice, discourse and explanations. In *Proceedings of the third Alvey Explanation Workshop*, 1987.

- [Gilbert 87b] G.N. Gilbert. Cognitive and social models of the user. In *Interact 87*, pages 165–169, 1987.
- [Goodyear 88] Peter Goodyear. Approaches to the empirical derivation of teaching knowledge for intelligent tutoring systems. In *Proceedings of the International Conference on Intelligent Tutoring Systems*, 1988.
- [Grice 75] H.P. Grice. Logic and conversation. In P. Cole and J. Morgan, editors, *Syntax and Semantics III - Speech Acts*, pages 41–58, New York: Academic Press, 1975.
- [Grosz & Sidner 85] B.J. Grosz and C.L. Sidner. *The Structures of Discourse Structure*. Technical Report 6097, Bolt Beranek and Newman, 1985.
- [Grosz 77] B. Grosz. *The Representation and Use of Focus in Dialogue Understanding*. Technical Report 151, SRI International, 1977.
- [Hagglund 81] S. Hagglund. Dialogue models for man-machine communication. A practitioners view. In *Proceedings of the Workshop on Models of Dialogue*, pages 47–58, 1981.
- [Halliday & Hasan 76] M.A.K. Halliday and R. Hasan. *Cohesion in English*. Longman, 1976.
- [Halliday 73] M.A.K. Halliday. *Explorations in the Functions of Language*. Edward Arnold, 1973.
- [Harris 88] Sandra Harris. Court discourse as genre: some problems and issues. In R.P. Fawcett, editor, *New Directions in Systemic Linguistics : Theory and Application*, pages 94–115, Francis Pinter, 1988.
- [Hartley & Smith 88] R. Hartley and M. Smith. Question answering and explanation giving in on-line help systems. In John Self, editor, *Artificial Intelligence and Human Learning*, pages 338–360, Chapman and Hall, 1988.

- [Hayes & Reddy 83] P.J. Hayes and D.R. Reddy. Steps towards graceful interaction in spoken and written man-machine communication. *International Journal of Man Machine Studies*, 19, 1983.
- [Hobbs & Evans 79] J. Hobbs and D. Evans. *Conversation as Planned Behaviour*. Technical Report 203, SRI International, 1979.
- [Hobbs 77] J. Hobbs. Coherence and interpretation in English text. In *Proceedings of the 5th International Conference on Artificial Intelligence*, pages 110–116, 1977.
- [Hollan & Hutchins 84] J.D. Hollan and E.L. Hutchins. Reservations about qualitative models. *Proceedings of the 6th Annual Conference of the Cognitive Science Society*, 1984.
- [Houghton & Isard 85] G. Houghton and S. Isard. Why to speak, what to say and how to say it : modelling language production in discourse. In *Proceedings of the International Workshop on Modelling Cognition*, University of Lancaster, 1985.
- [Hovy & McCoy 89] Eduard. H. Hovy and Kathleen F. McCoy. Focusing your rst: A step toward generating coherent multisentential text. In *Proceedings of the Conference of the Cognitive Science Society*, 1989.
- [Hovy 85] E.H. Hovy. Integrating text planning and production in generation. In *Proceedings of the 9th International Conference on Artificial Intelligence*, pages 848–851, 1985.
- [Hovy 88a] E.H. Hovy. *Planning Coherent Multisentential Text*. Technical Report ISI/RS-88-208, University of Southern California, 1988.
- [Hovy 88b] E.H. Hovy. *Two Types of Planning in Language Generation*. Technical Report ISI/RS-88-209, University of Southern California, 1988.
- [Hovy 89] Eduard Hovy. Unresolved issues in paragraph planning. 1989. (full paper) Submitted to proceedings of the European Workshop on Natural Language Generation.

- [Jameson 83] A. Jameson. Impression monitoring in evaluation-oriented dialogue. In *Proceedings of the 8th International Conference on Artificial Intelligence*, pages 616-620, 1983.
- [Jefferson 72] G. Jefferson. Side sequences. In D. Sudnow, editor, *Studies in Social Interaction*, Macmillan, 1972.
- [Johnson 87] A. Johnson. *Electronics: A Systems Approach*. Hodder and Stoughton, 1987.
- [Jullien & Marty 89] C. Jullien and J.C. Marty. Plan revision in person-machine dialogue. In *Proceedings of the European ACL Conference*, pages 153-160, 1989.
- [Kass & Finin 87] Robert Kass and Tim Finin. Rules for the implicit acquisition of knowledge about the user. In *Proceedings of the National Conference on Artificial Intelligence*, pages 295-300, 1987.
- [Kendon 83] A. Kendon. Gesture and speech : How they interact. In J.M. Wiemann and R.P. Harrison, editors, *Nonverbal Interaction*, SAGE, 1983.
- [Kidd 85] A.L. Kidd. The consultative role of an expert system. In P. Johnson and S. Cook, editors, *Proceedings of the Conference of the British Computer Society Human Computer Interaction Specialist Group*, pages 248-254, Cambridge, 1985.
- [Kieras 85] D.E. Kieras. Thematic processes in the comprehension of technical prose. In B.J. Britton and J.B. Black, editors, *Understanding Expository Text : A Theoretical and Practical Handbook for Analyzing Explanatory Text*, Lawrence Erlbaum, 1985.
- [Kukich 83] K. Kukich. Design of a knowledge based report generator. *ACL, 21st Annual meeting*, 145-150, 1983.
- [Labov & Fanshel 77] W. Labov and D. Fanshel. *Therapeutic Discourse: Psychotherapy as Conversation*. Academic Press, 1977.

- [Lesgold 88] A. Lesgold. Towards a theory of curriculum design for use in Intelligent Instructional Systems. In H. Mandl and A. Lesgold, editors, *Learning Issues for Intelligent Tutoring Systems*, pages 114–137, New York Springer Verlag, 1988.
- [Leudar & Antaki 87] Ivan Leudar and C. Antaki. Natural dialogue and human interaction: completion, closure and dynamics. In *Proceedings of the Second Alvey Explanation Workshop*, 1987.
- [Levin & Moore 77] J.A. Levin and J.A. Moore. Dialogue-games: metacommunication structures for natural language interaction. *Cognitive Science*, 1:395–420, 1977.
- [Levinson 85] S. L. Levinson. *Pragmatics*. Cambridge, 1985.
- [Linde 77] C. Linde. The organisation of discourse. In T. Shopen A. Zwicky P. Griffer, editor, *The English Language : English in its Social and Historical Context*, 1977.
- [Litman & Allen 84] D.L. Litman and J.F. Allen. A plan recognition model for clarification subdialogues. In *Proceedings of COLING '84*, 1984.
- [Litman 85] D.L. Litman. *Plan Recognition and Discourse Analysis: An Integrated Approach for Understanding Dialogues*. Technical Report 170, University of Rochester, 1985.
- [Macmillan & Sleeman 87] S.A. Macmillan and D.H. Sleeman. An architecture for a self-improving planner for intelligent tutoring system s. *Computational Intelligence*, 1987.
- [Macmillan et al 88] S.A. Macmillan, D. Emme, and M. Berkowitz. Instructional planners: lessons learned. In J. Psotka, L.D. Murray, and S.A. Mutter, editors, *Intelligent Tutoring Systems: Lessons Learned*, pages 17–27, Lawrence Erlbaum, 1988.
- [Malcolm & Smithers 88] Chris Malcolm and Tim Smithers. *Programming assembly robots in terms of task achieving behavioural modules : First experimental results*. Research Paper 410, DAI, University of Edinburgh, 1988.

- [Mandler & Johnson 80] J.M. Mandler and N.S. Johnson. On throwing out the baby with the bathwater: A reply to black and wilensky's evaluation of story grammars. *Cognitive Science*, 4:305-312, 1980.
- [Mann & Moore 81] W.C. Mann and J.A. Moore. Computer generation of multiparagraph text. *American Journal of Computational Linguistics*, 7:17-29, Jan-Mar 1981.
- [Mann & Thompson 87] W.C. Mann and S.A. Thompson. *Rhetorical Structure Theory: A Framework for the Analysis of Texts*. Technical Report RS-87-190, USC/Information Sciences Institute, 1987.
- [Mann 87] W.C. Mann. *Text Generation: The Problem of Text Structure*. Technical Report ISI/RS-87-181, University of Southern California, 1987.
- [Mayer 85] R.E. Mayer. Structural analysis of science prose: Can we increase problem solving performance. In B.J. Britton and J.B. Black, editors, *Understanding Expository Text : A Theoretical and Practical Handbook for Analyzing Explanatory Text*, Lawrence Erlbaum, 1985.
- [McCoy & Cheng 88] K.F. McCoy and J. Cheng. Focus of attention: constraining what can be said next. In *Proceedings of the 4th International Workshop on Natural Language Generation*, 1988.
- [McCoy 84] Kathleen F. McCoy. Correcting object-related misconceptions: how should the system respond. *COLING '84*, 444-447, 1984.
- [McCoy 85] Kathleen F. McCoy. The role of perspective in replying to property misconceptions. In *Proceedings of the ninth International Joint Conference on Artificial Intelligence*, pages 791-193, 1985.
- [McCoy 86] Kathleen McCoy. The romper system: responding to object-related misconceptions using perspective. In *Proceedings of the 24th Annual Meeting of the ACL*, pages 97-105, 1986.
- [McDonald 83] D. McDonald. Description directed control: its importance for natural language generation.

- In *Readings in Natural Language Processing*, pages 519–537, Morgan Kaufmann, 1983.
- [McDonald 86] D.D. McDonald. *Transistion Networks for Discourse Management*. Technical Report, University of Massachusetts at Amherst, 1986.
- [McKeown & Swartout 88] K.R. McKeown and W.R. Swartout. Language generation and explanation. In *Advances in Natural Language Generation: An Interdisciplinary Perspective (Vol 1)*, Pinter, London, 1988.
- [McKeown 85] K.R. McKeown. *Text Generation : Using discourse strategies and focus constraints to generate natural language text*. Cambridge University Press, 1985.
- [McKeown 88] Kathleen McKeown. Generating goal-oriented explanations. *International Journal of Expert Systems Research and Applications*, 4:377–395, 1988.
- [McKeown et al 85] K.R. McKeown, M. Wish, and K. Matthews. Tailoring explanations for the user. In *Proceedings of the 9th International Conference on Artificial Intelligence*, pages 794–798, 1985.
- [Miyake 86] N. Miyake. Constructive interaction and the iterative process of understanding. *Cognitive Science*, 10:151–177, 1986.
- [Moore & Swartout 88a] Johanna D. Moore and William R. Swartout. *Explanation in Expert Systems: A Survey*. Technical Report ISI/RR-88-228, USC/ISI, December 1988.
- [Moore & Swartout 88b] Johanna D. Moore and W.R. Swartout. Planning and reacting. In *AAAI Workshop on Text Planning and Generation*, 1988.
- [Moore 89] Johanna D. Moore. Responding to 'huh?' : answering vaguely articulated follow-up questions. In *Proceedings of the 1989 Conference on Human Factors in Computing Systems*, 1989.
- [Moyse 89] Rob Moyse. Knowledge negotiation implies multiple viewpoints. In Dick Bierman, Joost

- Breuker, and Jacobijn Sandberg, editors, *Artificial Intelligence and Education*, pages 140–149, 1989.
- [Murray 89] Bill Murray. Control for intelligent tutoring systems: a blackboard-based dynamic instructional planner. In Dick Bierman, Joost Breuker, and Jacobijn Sandberg, editors, *Artificial Intelligence and Education*, pages 150–168, 1989.
- [Neal *et al* 88] J.G. Neal, Z. Dobes, K. Bettinger, and J.S. Byoun. Multi-modal references in human-computer dialogue. In *7th National Conference on Artificial Intelligence*, 1988.
- [O'Malley 87] Claire O'Malley. Applying studies of natural dialogue to human computer interaction : are two heads better than one. In *Proceedings of the Second Alvey Explanation Workshop*, 1987.
- [Paris 87] C.L. Paris. Combining discourse strategies to generate descriptions to users along a naive/expert spectrum. *Proceedings of the 10th International Joint Conference on Artificial Intelligence*, 626–632, 1987.
- [Paris 88] C. Paris. Tailoring object descriptions to a user's level of expertise. *Computational Linguistics (Special Issue of User Modelling)*, 14, September 1988.
- [Peachey & McCalla 86] D.R. Peachey and G.I. McCalla. Using planning techniques in intelligent tutoring systems. *International Journal of Man-Machine Studies*, 24, 1986.
- [Petrie-Brown 89] Alison Petrie-Brown. Intelligent tutoring dialogue: the structures of an interaction. In Dick Bierman, Joost Breuker, and Jacobijn Sandberg, editors, *Artificial Intelligence and Education*, pages 195–201, 1989.
- [Pollack *et al* 82] M.E. Pollack, J. Hirschberg, and B. Webber. User participation in the reasoning processes of expert systems. In *Proceedings of the National Conference on Artificial Intelligence*, pages 358–361, 1982.

- [Power 74] R. Power. *A Computer Model of Conversation*. Unpublished PhD thesis, School of Artificial Intelligence, University of Edinburgh, 1974.
- [Power 87] R. Power. Efficiency in conversation. In *Proceedings of the Second Alvey Explanation Workshop*, 1987.
- [Reichman 81] R. Reichman. *Plain Speaking: A Theory and Grammar of Spontaneous Discourse*. Technical Report 4681, Bolt Beranek and Newman, 1981.
- [Reilly et al 88] R. Reilly, G. Ferrari, and I. Prodanof. Framework for a model of dialogue. In *Proceedings of COLING '88*, pages 540-543, 1988.
- [Rich 83] E. Rich. Users are individuals : individualising user models. *International Journal of Man-Machine Studies*, 18, 1983.
- [Ringle & Bruce 81] M.H. Ringle and B.C. Bruce. Conversation failure. In W.G. Lehnert and M.H. Ringle, editors, *Strategies for Natural Language Processing*, Lawrence Earlbaum, Hillsdale, New Jersey, 1981.
- [Rumelhart 75] D.E. Rumelhart. Notes on a schema for stories. In D. Bobrow and A. Collins, editors, *Representation and Understanding*, pages 211-238, Academic Press, 1975.
- [Russel 88] D.M. Russel. IDE: The interpreter. In J. Psotka, L.D. Murray, and S.A. Mutter, editors, *Intelligent Tutoring Systems: Lessons Learned*, Lawrence Erlbaum, 1988.
- [Rymaszewski 87] R. Rymaszewski. Negotiating cooperative effective explanations. In *Proceedings of the Second Alvey Explanation Workshop*, 1987.
- [Sable et al 85] J.D. Sable, R.M. Peterson, and A.D. Robbi. Skipper: a prototype sailing instructor. In *Proceedings of the IEEE conference on Applications of Artificial Intelligence*, 1985.
- [Sacerdoti 77] E.D. Sacerdoti. *A Structure for Plans and Behaviour*. Elsevier North Holland, 1977.

- [Sacks *et al* 78] H. Sacks, E.A. Schegloff, and G. Jefferson. A simplest systematics for the organisation of turn taking in for conversation. In J.N. Schenkein, editor, *Studies in Organisation of Conversational Interaction (Vol 1)*, pages 7-55, New York : Academic Press, 1978.
- [Schmauks & Reithinger 88] D. Schmauks and N. Reithinger. Generating multimodal output - conditions, advantages and problems. In *COLING '88*, pages 584-588, 1988.
- [Schuster 88] E. Schuster. Establishing the relationship between discourse models and user models. *Computational Linguistics (Special Issue on User Modelling)*, 14, September 1988.
- [Schuster 83] E. Schuster. *Understanding Misunderstandings*. Technical Report MS-CIS-82-679, University of Pennsylvania, 1983.
- [Scott *et al* 84] A. Carlisle Scott, W.J. Clancey, R. Davis, and E.H. Shortliffe. Methods for generating explanations. In Buchanan B.G. and Shortliffe, editors, *Rule Based Expert Systems*, Addison Wesley, 1984.
- [Searle 69] J.R. Searle. *Speech Acts: An Essay in the Philosophy of Language*. Cambridge University Press, 1969.
- [Simmons 85] R. Simmons. Self explanatory device descriptions. *SIGART Newsletter*, 93:24-25, 1985.
- [Sinclair & Coulthard 75] J. McH. Sinclair and R.M. Coulthard. *Towards an Analysis of Discourse : The English used by teachers and pupils*. Oxford University Press, 1975.
- [Sleeman 85] D. Sleeman. UMFE: A user modelling front end subsystem. *International Journal of Man-Machine Studies*, 23:71-88, 1985.
- [Stevens & Collins 77] A.L. Stevens and A. Collins. *The Goal Structure of a Socratic Tutor*. Technical Report 3518, Bolt, Beranek and Newman, 1977.

- [Stevens & Steinberg 81] A. Stevens and C. Steinberg. *A Typology of Explanations and its Application to Intelligent Computer Aided Instruction*. Technical Report 4626, Bolt, Beranek and Newman, 1981.
- [Stubbs 76] M.J. Stubbs. Keeping in touch: some functions of teacher-talk. In M. Stubbs and S. Delamont, editors, *Explorations in Classroom Observation*, Wiley, 1976.
- [Stubbs 83] M. Stubbs. *Discourse Analysis : The Sociolinguistic Analysis of Natural Language*. Blackwell, 1983.
- [Suchman 87] L. Suchman. *Plans and Situated Actions*. Cambridge University Press, 1987.
- [Swartout 83] W.R. Swartout. *XPLAIN: A System for Creating and Explaining Expert Consulting Programs*. Technical Report ISI-RR-83-4, Information Sciences Institute, University of Southern California, 1983.
- [vanBeek 86] P.G. van Beek. *A Model for User Specific Explanations from Expert Systems*. Technical Report CS-86-42, Dept. Computer Science, University of Waterloo, 1986.
- [vanDijk 85] Teun A. van Dijk. Structures of news in the press. In Teun A. van Dijk, editor, *Discourse and Communication : New Approaches to the Analysis of Mass Media Discourse and Communication*, de Gruyter, 1985.
- [VanLehn 83] K. VanLehn. Human procedural skill acquisition: Validating an AI based theory. In *Proceedings of AAAI*, 1983.
- [Ventola 88] Eija Ventola. Text analysis in operation: a multilevel approach. In R.P. Fawcett, editor, *New Directions in Systemic Linguistics : Theory and Application*, pages 52-77, Francis Pinter, 1988.
- [Wachtel 86] T. Wachtel. Pragmatic sensitivity in natural language interfaces and the structure of conversation. In *Proceedings of COLING '86*, pages 35-41, 1986.

- [Wahlster & Kobsa 86] W. Wahlster and A. Kobsa. Dialogue based user models. *IEE Special Issue on NL processing*, 1986.
- [Wallis & Shortliffe 84] J.W. Wallis and E.H. Shortliffe. Customised explanations using causal knowledge. In Buchanan B.G. and Shortliffe, editors, *Rule Based Expert Systems*, Addison Wesley, 1984.
- [Weiner 79] J.L. Weiner. *The Structure of Natural Explanation: Theory and Application*. Technical Report SP-4035, Systems Development Corporation, 1979.
- [Weiner 80] J.L. Weiner. Blah, a system which explains its reasoning. *Artificial Intelligence*, 15:19-48, 1980.
- [Wenger 87] E. Wenger. *Artificial Intelligence and Education*. Morgan Kaufmann, 1987.
- [White & Frederiksen 85] B.Y. White and J.R. Frederiksen. Quest: qualitative understanding of electrical system trouble shooting. *SIGART Newsletter*, 93:34-37, 1985.
- [Winkels et al 88] Radboud Winkels, Joost Breuker, and Jacobijn Sandberg. Didactic discourse in intelligent help systems. In *Proceedings of the International Conference on Intelligent Tutoring Systems*, pages 279-285, 1988.
- [Wittgenstein 68] L. Wittgenstein. *Philosophical Investigations*. Oxford Blackwell, 1968.
- [Woolf & McDonald 84] B. Woolf and D.D. McDonald. Context-dependent transitions in discourse. *Proceedings of the National Conference on Artificial Intelligence*, 355-361, 1984.
- [Woolf & Murray 87] Beverly Woolf and Tom Murray. A framework for representing tutorial discourse. In *Proceedings of the 10th International Conference on Artificial Intelligence*, pages 189-192, 1987.
- [Young & Simon 87] R.M. Young and T. Simon. Planning in the context of human-computer interaction. In *Proceedings of HCI '87*, 1987.

[Zuckerman & Pearl 86]

I. Zuckerman and J. Pearl. Comprehension-driven generation of meta-technical utterances in math tutoring. In *Proceedings of the the 5th National Conference on Artificial Intelligence*, pages 606–611, 1986.

Appendix A

Human Explanations of Circuits

This appendix will include a selection of complete human explanations of circuit behaviour, some with partial analyses of the explanation. It will start with a set of simple uninterrupted explanations so that the structure of such explanations is clear, then move on to five explanatory dialogues. All the examples were transcribed without the self corrections and hesitations which are part of all conversation — we are interested in expert performance. The examples selected are explanations which appeared clear and where the explainer did not get too confused. Text book explanations will be included for comparison.

The circuit diagrams used in the explanations are given at the end of the appendix.

A.1 Uninterrupted Explanations

These examples were given by a senior lecturer in physics at the local teacher training college, and are especially clear.

Light Detector

- 1) Well, this here looks to be a potential divider circuit
- 2) and it contains two components
- 3) one of which is a light dependent resistor
- 4) and the other is a variable resistor.
- 5) The purpose of this circuit is to provide a varying output voltage on this line here.
- 6) The idea is that when light falls on this light dependent resistor its resistance changes,
- 7) so therefore because the resistance of this series circuit has changed the voltage at this point here can change.
- 8) So, when light falls on the LDR the resistance of this part of the

circuit is very low

9) and so this voltage here is a high voltage.

10) In darkness the resistance of the LDR is large

11) and so the voltage at this point drops to a low value.

12) So, we can get a high or low output depending on how much light is falling on the light dependent resistor.

The content of this example was examined in chapter 3, and classified as :

- Identification. *This is a potential divider circuit.*
- Constituency. *And it contains variable resistor.*
- Function. *The purpose*
- Process. Lines 6-11 (consisting of 'cause-effect' propositions giving the causal process for different sorts of inputs).
- Conclusion (behaviour). *So we can get*

The process section could be further divided into a trace of what depends on what, process when light falls and process in darkness.

There are few discourse features worth noting, though some discourse markers ('well' and 'so') are used, probably to signal sub-topic boundaries or conclusions.

The next example illustrates the explanation of a similar (analogous) device:

Heat Detector

1) Well, this is similar to the one above except instead of a light dependent resistor, we've got a thermistor

2) which is a device whose resistance varies with temperature,

3) so as the temperature rises, then so does the resistance of the thermistor, or is it the other way round. There's a variation in temperature anyway in the thermistor.

4) And so this component will vary with temperature,

5) and so the voltage at this point here will vary also.

6) So, depending on how high or low the temperature is, this will determine whether we have a high or low output at this line here.

This can be classified as starting with an analogical description (**comparison** with the analogical device and **cause-effect** of the new component), followed by **process** and **conclusion** (behaviour).

Inverter

- 1) Well, this is a circuit that has a transistor in it,
- 2) and assuming that this line here represents an output voltage,
- 3) its state, that is whether it's high or low, will depend on what voltage we have applied.
- 4) So, for example, in this case here, if a positive voltage is applied to that input the transistor will conduct,
- 5) so the voltage at the bottom of that resistor there becomes close to zero.
- 6) And so a high input provides a low output.
- 7) Conversely, a low input will provide a high output.
- 8) So this inverts the voltage.

This example, again, can be classified as: **constituency** (1-3), **process** (4-5), **conclusion** (cause-effect) (6-8). Of course, finer categorisation is possible of each section. Constituency here consists of the identification and behaviour of the new component, process as a trace for high input voltage and conclusion the behaviour for both high and low voltages and its functional behaviour. These finer details of structure are captured in the explanation grammar given in chapter 3.

The final example explanation from this subject is given below, this time for a slightly more complex circuit:

Temperature Warning

- 1) This is a circuit which contains two transistor, and a potential divider setup again.
- 2) So the idea here with the thermistor is that as the temperature changes the input voltage to the base of the transistor varies also.
- 3) So it will be either high or low depending on the ambient temperature.
- 4) Well, let's assume for the moment that the voltage to the base of the transistor goes high.
- 5) This would mean that the voltage at the collector of transistor T1 is going to be low,
- 6) which will mean that transistor T2 is not conducting
- 7) so the lamp will not light.
- 8) If it is the other way round so that the transistor is shut off, the voltage here is high,
- 9) which means that the transistor is now going to conduct
- 10) which means that the lamp will light.

This example is classified as: **constituency** (1-3), **process** (4-10). The constituency section consists of identification of components and description of

behaviour of one section. The process section explains behaviour at a slightly higher level than the smaller circuits, missing out steps where they are clear.

The final example in this section will be an explanation of the heat warning circuit by another expert, to illustrate at least some consistency across explainers.

Heat Warning

- 1) This is a circuit with a lamp, and a thermistor.
- 2) It's got two transistors which look like they are a two stage inverting amplifier.
- 3) So as the temperature of the thermistor varies the brightness of the bulb will vary as well.
- 4) There's also the potentiometer so the range of brightness can effectively be varied as well.
- 5) So, as the temperature goes up I think the resistance goes down on the thermistor.
- 6) Which means that transistor T1 will be turned on more
- 7) and the voltage at the collector of T2 will go down
- 8) and this will turn off T2
- 9) and make the bulb go dimmer.
- 10) So it looks like the light will come on when it gets cold.

This begins with a complex **constituency** description including the function of the components within the circuit (1-4). This is followed with a single **process** description (4-9) followed by a **conclusion** (single behaviour description) in line 10.

So, these five examples illustrate common patterns of description across a range of circuits and two experts. More detailed analysis revealed the more complex range of structures given in chapter 3. There are of course many examples which are less coherent, often involving just a hesitant process sequence. But the ones chosen above capture much of the richness of this sort of explanations, where the expert has a good understanding of the domain.

A.2 Text Book Explanations

Text book explanations of the same circuits are very similar to the clearer verbal explanations, but may include additional information related to the context of the explanation (why this explanation is being given at this point in the text-book) — the artificial setting of the verbal explanations effectively eliminated this context dependent information. They may also miss out information introduced in a previous section — for example, the two circuits below were identified in the preceding section of the text book as potential divider circuits with output

voltage depending on input heat/light intensity. The following two examples are taken from [Johnson 87] and follow a discussion of potential divider circuits:

Light Detector

“This is a basic circuit consisting of a transducer and a resistor as described in the previous section (Figure 4.15). The transducer is an LDR (section 3.3) and it forms the upper half of the potential divider. If the intensity of the light falling upon the LDR increases, the resistance of the LDR will fall. The output voltage from the potential divider will therefore rise.

The potential divider will give the greatest change in output voltage when the resistances of the LDR and lower resistor are equal, and the resistor is made variable so that this can be achieved. By adjusting the potentiometer it is possible to operate the circuit successfully over a wide range of light intensities.”

Heat Detector

“Once again a potential divider is used with a transducer as the upper resistor, but this time the transducer is a thermistor (section 3.2). As the temperature of the thermistor rises so too does the output voltage of the circuit (See figure 4.16). The resistor is a potentiometer connected as a variable resistor to allow for different temperature ranges.

In spite of the similarities in their circuits, the light unit and the heat unit are very different to use as parts of a system. The resistance of an LDR varies from nearly infinity in the dark to a few hundred ohms in a good light. When the LDR resistance is infinite, the output voltage will be close to zero, and when it is at its lowest the voltage will approach V_{cc} . This very large range in output voltages makes light operated systems easy to design.

The resistance of a thermistor falls by no more than a factor of ten, which produces an output voltage change of only a few volts. This makes the design of the next stage in the system much more crucial.”

Note that even these written explanations are interactive — If the reader does not remember what a thermistor is, the reference is given to look this up.

A.3 Dialogues

This section will give a number of examples of complete dialogues with novices about circuit behaviour. They are chosen to illustrate a range of interactional styles, and extend the example dialogues in chapter 3 to complete explanations of a particular circuit. The first two examples are with more or less complete novices, resulting in largely unstructured dialogues dominated by the novice's questions. All the dialogues are by different subjects.

A.3.1 Followup Questions

The first example was introduced in figure 3-7. It illustrates a dialogue which is dominated by followup questions, following a very short but coherent explanation from the expert who misjudges the novice's knowledge. In this extreme example, many of the followup answers are never completely given — they are interrupted by more basic questions. The example illustrates the problems and confusions which may arise in a simple explanatory dialogue when the novice does not have the requisite knowledge — and therefore the usefulness of at least having some model of what the user knows, learnt from the interaction or direct questions. The EDGE model does not address the issue of dealing with followup questions in any detail, though it does allow for followup questions at the end of an explanation.

Light Detector

- 1) In this circuit we have an output whose voltage depends on the amount of light falling on the LDR
- 2) and the more light falls on here the more it will conduct
- 3) so the higher the voltage will be here.
- 4) — **What does LDR stand for?**
- 5) Light dependent resistor.
- 6) — **What's this bit?**
- 7) That bit's a variable resistor so you can change how the output level varies with the resistance of the LDR.
- 8) — **Say that again..**
- 9) Well, this is a potential divider here, so the potential here will be between zero and 9V in proportion to the resistance of these two things.
- 10) — **What's potential?**
- 11) The voltage.
- 12) — **Why do you call it potential?**
- 13) Because that's what trendy people call it.
- 14) — **and this is a variable resistance?**

- 15) Yes, that's right.
- 16) — **Why does it say 5K on it?**
- 17) Because its maximum resistance is 5000 Ohms.
- 18) — **Right. Does it matter where these things are?**
- 19) Yes, if you switch them round it will work the other way round, and the voltage will go down as the light increases.
- 20) — **Why?**
- 21) Because, the ratio, the output will vary between 0 and 9V in proportion to the ratio of the two resistances,
- 22) so the bigger the resistance here, the further it will be away from 0V.
- 23) — **Is there any way you can find out what the resistance is in that, at any given point?**
- 24) Well, if you know what the resistance of this is, you know what the output voltage is
- 25) then you know that the ratio of the resistance of the light dependent resistor to the variable resistor here is the same as the ratio of the voltage difference between 9V and the output and the output and 0V.
- 26) — **The output comes out here? What comes outhere?**
- 27) That's where your 9V goes in, you connect these to your battery.
- 28) — **I see, that's why it says plus there..**
- 29) This is at 0V and this is at +9V. It says plus because it's a positive voltage. It could be -9V.
- 30) — **So what's happening here?**
- 31) You could just choose an arbitrary.. voltages are relative things, you can choose an arbitrary voltage as you 0V point.
- 32) — **Why is there a different voltage in different places?**
- 33) Because there's a battery attached to it, which they don't show. So when they do this they really mean there's a battery here. (*draws battery*). It's a 9V battery.
- 34) — **And this is the positive pole and this is the..**
- 35) Just so, so this one is 9V higher than this one. We could cross this out and say -300V here and this would be -291V. All you've got is the difference between them is 9V.
- 36) — **Alright.**
- 37) Enough of that one.

A.3.2 Interrupted Explanations

The next example (introduced in figure 3-8) illustrates a case where the expert does apparently attempt to get back to the planned explanation. In this case the explanation is given in terms of circuit equations rather than systems, but the same pattern of explanations applies. This example will be marked according

to the authors perception of where interruptions begin and end, marking interrupting sequences with a '#'. The example also illustrates the importance of the novice's signals when they understand.

Light Detector

1) These components here, you might consider them as being both resistors. Two variable resistors. I can write down a relation for resistance..

2) # — **You'll have to tell me what a resistor is.**

3) # A resistor is an element which if you apply a voltage across it there is a certain current flowing through the resistor. There is a relationship $V=I \cdot R$..

4) # — **It doesn't help. What does it do? Does it stop the current getting through?**

5) # Yes. If the resistance was nil it would just be a connection here, then the current is mainly restricted by the amount of current.. voltage..

6) # — **Right, so the resistor has to be, can be of various strengths presumably. And the current going through it has to be of certain strengths before it can get through the resistor?**

7) # No, in this case the current is determined by the strength of your power supply and you can rewrite this relation to $I=V/R$, where if this is constant, say 10V or 9V battery for example, changing this resistor changes the current through the resistor. If this one becomes nil you get an infinite current. If it's high then the current becomes low.

8) # — **I see, or at least I think I see.**

9) Well, this circuit here are just two resistors, switched in serial mode. You can think of it as one resistor supplied with a voltage 9V, and there will be a current flowing through the circuit.

10) Well, the output can be calculated by, when you know the current flowing through here then you assume that the output voltage is determined by the current multiplied by this resistance. In order to calculate the current you have to include this resistor in your network as well..

11) # — **Why are we bothered about what the current is through there then? What's going to happen? I have to know the purpose of it.**

12) # In the end you want to know this voltage. The output voltage here relates to the input voltage, and your current can be assumed as constant, not as constant, as variable depending on the value of the resistance.

13) # — **OK**

14) I can just write down a relationship, this is I_1 , the output voltage

is I_1 / (*writes equation*).

15) — Right

16) Light on the resistor - this is a light sensitive resistor - changes this value, your resistance value of this element changes, so in the end your output voltage changes as well.

17) # — Why would that change naturally, in daylight?

18) *clarification from experimenter*

19) # — A sudden change in light will change that, so what happens? This is a very kind of low level description, and without knowing why it's worthwhile having your voltage be anything, I mean, why don't you just plug it into the mains?

20) # Maybe I better explain it on a higher level.

21) # — I mean, what is the point of this particular bit, that is really what I want to know, rather than an explanation at a lower level.

22) # OK, well, you want to have varying output signal depending on the light falling on this input here, and you also want to set the sensitivity of this element as well, and that's done by this varying resistor. The amount of light, there is a direct relationship between the change in voltage here and the light falling on there, but the level on which your voltage changes can be adjusted by this resistor. So it will change maybe 10% at 5V, when this one is set, depending on the light falling, or you can adjust the threshold a little bit so it's 6V or 7V.

23) # — So you can make it more or less sensitive, I mean, this one must be sensitive to what the LDR is doing?

24) # I think sensitivity is a property of the LDR itself.

25) # — OK

26) Is that sufficient?

27) — I think I know what's going on with it, with the first bit.

This example isn't a particular clear explanation. It follows the basic 'constituency-process' type description in lines 1, 9-10 and 16, but confuses the novice with unnecessary details. However, it is a rich source of examples of how interruptions are managed. The end of an interruption is signalled by the novice with 'OK' or 'I see', and then the expert apparently continues the main explanation, in one case repeating part of the previous explanation.

The novice gives some acknowledgement that she understands in three places — when she has grasped the last utterance (line 15), when she accepts the answer to a question (lines 8, 13, 25) and when she has understood the circuit (line 27). All these are important for the novice to understand, and for the expert to perceive any problems.

A.3.3 Guided Explanations

This example shows a dialogue where the expert dominates, but involves the novice by asking directed questions. An extract was given in figure 3-9.

Inverter

- 1) OK, do you remember anything about transistor?
- 2) Basically, the important thing about transistors is, you've got three ends to it, the middle one, that's called the base, and current going through the other two, and also a little bit going through the base. And typically you get a little current going through the base and a big current going through the other two, and the current you get going through the base controls the current through the other two.
- 3) In other words, you can think of it as a varying resistor between the two, which, when there's a lot of current going through here, the resistance is low so a lot of current goes through these two. And when there's no current going through the base, there's no current going through these two
- 4) and it acts as an amplifier, so the current going through the collector and emitter
- 5) — is amplified?
- 6) is 100 or 200 times bigger than the current going through here.
- 7) The arrow here just tells you which way round you should connect it.
- 8) Right, so what's going to happen here, say, if you consider this as an input and this as an output, what happens when you make this voltage high, what happens to this?
- 9) — When you make it high you probably get all the current from this end amplified.
- 10) Well, when you make the current high..
- 11) — It opens the gate.
- 12) What's going to happen to the resistance between here and here?
- 13) — It's going to be low.
- 14) So, what's going to happen to the voltage here. If you have a very low resistance?
- 15) — It's going to tend to the high limit.
- 16) Careful, if you let the current through these two, then it's just as if you wired these two up, a little bit, so it's just as if the transistor weren't there and these two are connected.
- 17) So, if these two are connected there's nothing here in fact. What's the voltage here going to be?
- 18) — Well, the same voltage as the other end.
- 19) Yes, so it's going to be low.
- 20) So, when this is high this is low

- 21) and on the other hand, when the voltage here is low, there will be no current going through here, so what will the voltage be?
- 22) — High.
- 23) Yes, high, 9V.

This explanation starts with a **constituency** description (1-7) — a detailed description of the main component (transistor). Lines 8-19 are a detailed **process** description with a summary of this behaviour in line 20. Lines 21-23 conclude with the process description for the other input value considered. The explanation is given at a more detailed level than the ones above, but follows much the same format.

The more interesting issues in this explanation relate to the interactions with the novice. The novice is being asked to fill in some of the steps in the explanation — but these steps follow the basic structure of the explanation. When the novice gets questions wrong there are various strategies adopted for correcting the user - this expert never immediately told the novice the correct answer following an error. Lines 10-19 illustrate a complex remedial sequence, where each incorrect or partially correct answer leads to further information and further questioning, leading the novice to the correct answer. Lines 16-17 illustrate a basic strategy used in the EDGE system — if the answer is incorrect, give them some more information from which they should be able to deduce the answer and ask it again.

A.3.4 Explanations Dominated by the Novice

This example, introduced in figure 3-10, shows how an explanation may be dominated by the novice, with the expert commenting. The first few utterances (1-4) and lines 7-10 again illustrate how an explanation is resumed after an interruption. In the second case it was an interruption from the expert after an incorrect statement from the novice.

Heat Detector

- 1) Right, the next thing is exactly analogous really..
- 2) # OK, what's this?
- 3) # That means it's a variable resistor, it's a resistance that you can vary by turning that little knob, effectively, and it varies from 5000 Ohms, thats a fairly high resistance to presumably nothing.
- 4) Right, the heat detector is exactly analogous in fact.
- 5) How do you think it works?
- 6) Right, presumably depending on the amount of heat here, if it has, the more heat it has the more resistance it has?
- 7) # Probably the less.
- 8) # The less heat it has the more resistance it has?

- 9) # Sorry, no, the hotter it is the less resistance.
- 10) OK, the hotter it is the less resistance.
- 11) So, if it's not hot at all, if it's cold, if it's very high resistance therefore this middle one is closer to zero, and if it's very very hot it has low resistance so it's closer to 9V here.
- 12) # But I don't see what difference this 5K or 10K makes?
- 13) # It's probably because a thermistor has a high intrinsic, if the relative resistances mean that, decide whether it's going to be closer to 9 or 0, if this one's low relative to that one then it's closer to zero and if this one's low relative to that one it's 9. And presumably it's variable to allow you to tweak it slightly so you can adjust the point at which it swings from 9 to 0 exactly.
- 14) OK

A.3.5 Complex Cooperative Explanation

This final example shows how a complex circuit is explained jointly by expert and novice.

Burglar Alarm

- 1) What I'm going to do is to get you to explain this last circuit to me. Before I do that I better say briefly what a comparator is.
- 2) What this does is compare the inputs and if there's difference it should be switched on, if there's no difference it won't be switched on. I think it's that way round.
- 3) I'll tell you what we want from this burglar alarm and you'll try and explain how it can be done.
- 4) Now, we've got a speaker and what we want that to do is when a light gets shined on there we want that to go dee dah dee dah. Like that.
- 5) Now, you know what an astable circuit does. You know how the voltage can vary across here, you know that this compares voltages, comparing the voltage across there to the voltage coming in there. And we've got these two things which are — Can you remember what that circuit represents?
- 6) — It's called a NAND
- 7) Right, and you know that that's an inverter as well.
- 8) — So, when that goes, when it's off, that shouldn't be working, so I assume it's all happy at the moment, so that's at 9, that's at zero.
- 9) So when there's no light that will tell you that - what resistance, low or high?
- 10) — When there's no light that will be low.

- 11) High, high resistance.
- 12) — Didn't you say that when you shone light on it it got higher?
- 13) It gets lower.
- 14) — OK, so it's high, so this is 0, so that's zero and that's 9 so that won't be on cos that's at zero.
- 15) If I've got it right, that will be off, if either are zero that's going to be a one, so that's zero.
- 16) — You don't need to worry about this side of the circuit, it's just producing a block wave making the nee naw nee naw sound. So that's zero then this is off, which means that that's on, this is going on-off all the time, and so that would go nee naw nee naw.
- 17) Hold on. That's off, that's off, so the potential is at 9V, so that's at 9V, and going ee aw ee aw. Hold on..
- 18) — This should be one, so that's one, so that's off, yeah, the transistor is on, but it breaks the circuit, so this is all at 9V.
- 19) We're talking about a point up here. If that's a NAND, if either of these are off then that point is at 9 up there.
- 20) — Hang on, when we say off do we mean that..
- 21) Hold on, that's on, that's not what we wanted.
- 22) — Should it be zero?
- 23) *Clarification from experimenter*
- 24) — So, the circuit should now be on because the light beam has been broken. Let's go through it. This is one, it doesn't really matter what one is, this will be open, 1, so that will be at 9V. Each time this flicks on or off this will flick on or off and that would go dddddd. If you actually had light on there then..
- 25) That's not true, this will be switching on or off. What we want is this to be switched off if either of these two is off, and that is always going to be at 9. It doesn't matter whether that's changing or not, that can be zero or 1, so we actually want that to be zero. If that's a high voltage we get an inverter and that will be switched on.
- 26) *clarification*
- 27) — It would be an inverter if there was a wire coming out here.
- 28) Oh, right, but we'll have, that would be raised to zero volts.
- 29) — Right, if there's light shining on this, the resistance goes down, which means that this will get to 9, and that's already at 9, so this will be zero, that will be zero and because we have a zero here this is..
- 30) Up there, that thing there.
- 31) — We have a zero on the first one.

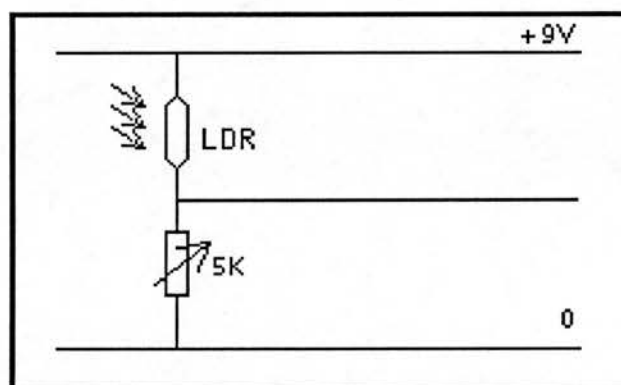
- 32) Well, the first one will be going from zero to 1 all the time.
- 33) — So, looking at th table, that point there, for this whole inverter will be going 1 1, so it will always be on.. 34) *clarification*
- 35) — So we've got one, so this is always going to be 9, so there's always voltage across there.
- 36) So it's not going to produce a sound. So this is one, where if you break the light is will..
- 37) — So that's going to happen in that case, and if it's the other case, where there's no light shining on here, this will be zero, this will be at 1, that'll be at 1, so if we draw the table again, that's at zero and 1 because that will be going up and down, this will be at zero and this one will be at 1, so this will be varying between 9 and 0 so will be gong nee naw nee naw.

Note the use of opening meta-comments in lines 1 and 3, and the fact that the circuit section so far (in the sequence of explanations) unexplained is introduced here (the comparator). This suggests that modelling the user is of some use. The explanation starts off with **function** (line 4), **consituency** (lines 2, 4-7) and **process** (line 8 onwards). Much of the explanation consists of repeated attempts by each subject (novice and expert) to get the process sequence right, with occasional clarifications from the experimenter when both parties were confused. It perhaps shows how collaborative explanations can fail to be constructive when a certain level of confusion is exceeded!

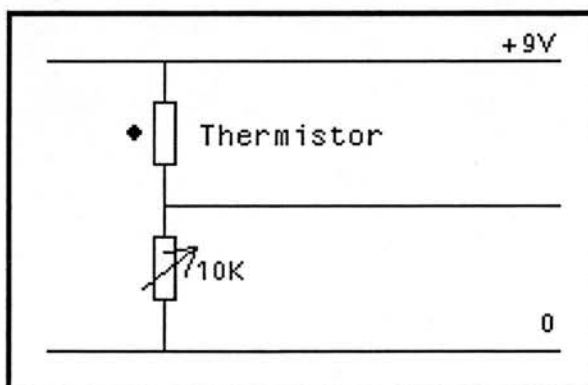
A.4 Circuit Diagrams

The above examples include explanations of five of the eight circuits used in the experiment. These five circuits are included below. The remaining three circuits included an astable multivibrator, a simple time delay circuit and a NAND gate.

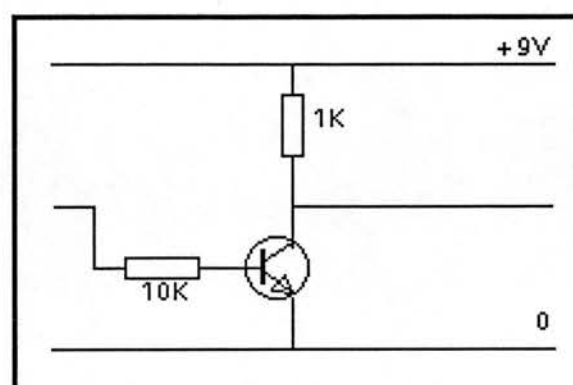
Light Detector Unit



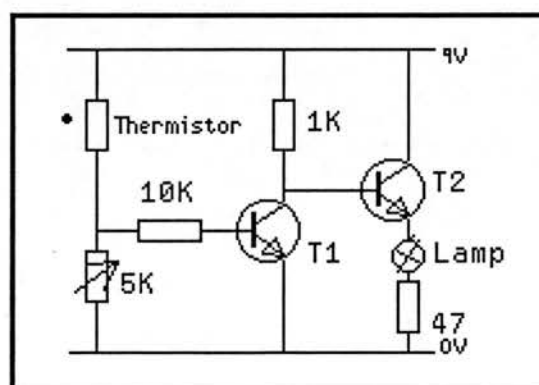
Heat Detector Unit



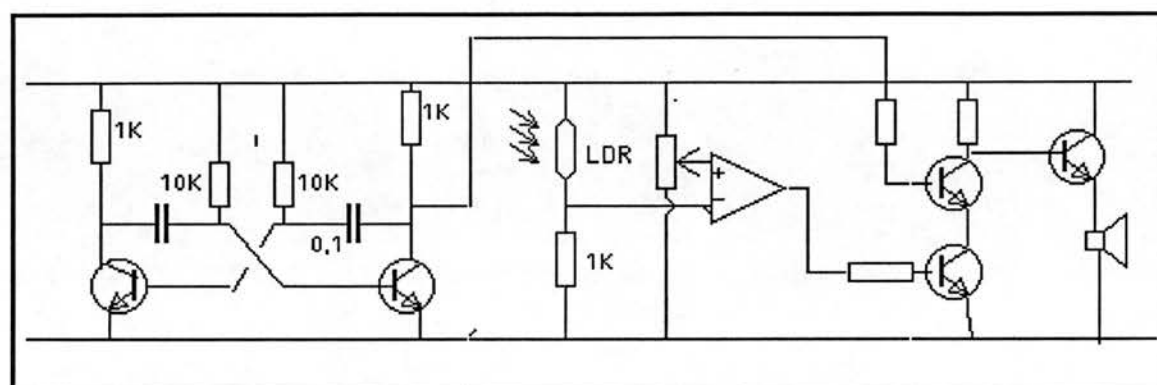
Inverter



Low Heat Warning Circuit



Burglar Alarm



Appendix B

Content Plans

Content plans are used to structure the domain content that is to be presented to the user. They are based on 'sub-skill' and prerequisite relationships between the concepts being taught, and knowledge of the structure of human explanations in this domain. The complete set of plans are included below, with comments on particular features¹.

The top level plan (figure 2-1) describes how to teach how something works. The plan requires that the device's structure is available in the domain models, and there is a precondition that the user knows the 'structure' of the device — this is defined to include the component behaviours. The body of the plan includes first initiating a sub-transaction on the sub-topic of giving the causal behaviour of the circuit, then initiating a concluding transaction describing what the circuit does. The 'plan' function is used as a shorthand for constructing a plan (record structure) with empty pointers to parent and sub-plans. The 'deviceref' function is a specialised version of the pronoun selection function — given a device instance such as 'light-detector0016' it will return either its name (eg "the light detector circuit") or the pronoun 'it' depending on the discourse context.

Next, the 'structure' plans illustrate how things can be explained in radically different ways depending on the user's knowledge (figure 2-2). Note that the template and difficulty levels apply to both versions of the plan. Unlike in the top level plan, sub transactions are not used. This decision is somewhat arbitrary based on a subjective view of where a topic shift requires marking². Decisions

¹Note that these plans are from the most up to date version of the system. There may be some slight variations from the system used in the evaluation, but these should not be great. This version was used to obtain the examples in the thesis.

²If the discourse is to be made more flexible, decisions about where to initiate sub transactions should be taken out of the plans themselves. However, the current system illustrates their use within a simple architecture.

```

(defplan how-it-works (device)
  :constraints ((getslot device 'structure))
  :preconditions ((structure device))
  :subgoals ((interaction sub.transaction
                        ((plan 'teach 'causal-behaviour (list device))
                         'mid))
              ((interaction sub.transaction
                        ((plan 'teach 'what-it-does (list device))
                         'conclusion))))
  :template ("how" (deviceref device) "works.")
  :difficulty (3 light-detector 2))

```

Figure B-1: Plan to explain how a device works

about how to decompose the skill are also somewhat tentative — perhaps ‘what-sort-of-device’ should be a precondition of the ‘what-its-for’ plan?

The ‘what-sort-of-device’ (figure 2-3) plan looks for a link with the user’s existing knowledge. It is obviously oversimple, as if it does not find one from the type of circuit or type of device it has no other strategies for finding links. However, it serves to illustrate this type of plan.

The two plans in figure 2-4 (giving a device’s function and drawing a simple analogy with another device) are again, very simple. There is much more to both device function and drawing analogies than these plans capture! However, they are sufficient in the restricted domain chosen to generate adequate explanations, and these sub-plans could be developed further if the plans were generalised.

Note the functions ‘compare-io’ and ‘compare-comps’. Because of the complexity of constructing even a simple analogy given the analogical devices, arbitrary lisp functions are used to extract the comparisons before propositions are constructed.

The plans in figure 2-5 identify and give the behaviour of all the unknown components in the circuit. This involves fairly complex plans calling graphical actions (figure 2-6). The ‘what-it-looks-like’ plan can only succeed if the user notices both the text and the graphics — the user model will be based on both graphical and textual actions. The ‘in/out’ plan involves considerable complexity in accessing input and output meters (windows) and their labels. Some of this could of course be evaluated in special functions, or as part of the text templates used — it would be reasonable, for example, to access the label of the meters/icon windows within the low level generation procedure.

The final plans in figure 2-7 link with the interpreter which calculates causal behaviour. This will in turn call teaching and discourse plans.

```

(defplan structure (device)
  :constraints ((getslot device 'analogy)
                (understands (list (getslot device 'analogy)
                                    'what-it-does T)))
  :subgoals ((teach similarity (device
                                (getslot device 'analogy))))
  :template ("what" (deviceref device)
             "is like (its structure and components)")
  :difficulty (2))

(defplan structure (device)
  :preconditions ((what-sort-of-device (device))
                 (what-its-for (device)))
  :subgoals ((teach components (device))))

```

Figure B-2: Describing a device

```

(defplan what-sort-of-device (device)
  :constraints ((getslot device 'circuit-type)
                (understands
                 (list (getslot device 'circuit-type)
                       'what-sort-of-device)))
  :subgoals ((interaction teaching.exchange
                        ((list 'circuit-type
                               device
                               (getslot device 'circuit-type)))))
  :template ("what sort of circuit" (deviceref device) "is."))

(defplan what-sort-of-device (device)
  :constraints ((getslot device 'device-type)
                (understands
                 (list (getslot device 'device-type)
                       'what-sort-of-device)))
  :subgoals ((interaction teaching.exchange
                        ((list 'device-type
                               device
                               (getslot device 'circuit-type)))))

```

Figure B-3: Linking to existing knowledge

```

(defplan what-its-for (device)
  :subgoals ((interaction teaching.exchange
    ((list 'purpose
      device
      (car (getslot device 'inputs))
      (getslot device 'outval))))))
  :template ("what" (deviceref device) "is for."))

(defplan similarity (device otherdevice)
  :subgoals ((interaction teaching.exchange
    ((list 'compare device otherdevice)))
    (interaction teaching.exchange
    ((list 'compare-io
      (compare-io device otherdevice))))
    (interaction teaching.exchange
    ((list 'compare-comps
      (comparecomps device otherdevice))))
    (teach components (device)))
  :template ("what the difference between the"
    (getslot device 'name) "and the"
    (getslot otherdevice 'name) "is."))

```

Figure B-4: Describing a Device

```

(defplan components (device)
  :constraints ((getcomps device))
  :subgoals ((foreach comp in (getcomps device)
    (if (greaterp (importancelevel
      (list comp 'what-it-does))
        1)
        (teach component (comp))))))
  :template ("what" (deviceref device)
    "s components are like."))

(defplan component (device)
  :preconditions ((what-it-looks-like (device)))
  :subgoals ((if (not (understands
    (list device) 'what-it-does))
    (interaction sub.transaction
      ((plan 'teach 'what-it-does (list device))
        'sub))))))
  :template ("what" (deviceref device) "is like"))

```

Figure B-5: Describing Components


```

(defplan what-it-looks-like (device)
  :subgoals ((call pointat (device))
             (interaction teaching.exchange
              ((list 'describe-comp device))))
  :template ("what" (deviceref device) "looks like."))

(defplan what-it-does (device)
  :subgoals ((foreach input in (getslot device 'inputs)
                       (teach in/out
                            (device input
                              (apply (getslot device 'runfn) input))))))
  :template ("what" (deviceref device) "does."))

(defplan in/out (device inputs output)
  :subgoals ((call clearvalues)
             (foreach input window
                      in inputs (getslot device 'inwindows)
                      (call putresult (input window)))
             (interaction teaching.exchange
              ((list 'describe-event device inputs output
                    (gettitle (getslot device 'outwindow))
                    (mapcar (getslot device 'inwindows)
                            (function (lambda (win)
                                         (gettitle win))))))))
             (call putresult (output
                             (getslot device 'outwindow)))
             (call pupil-acknowledge.act))
  :template ("what the output of" (deviceref device)
            "is when its inputs are:" inputs))

```

Figure B-6: What a component/device does

```

(defplan causal-behaviour (device)
  :constraints ((getslot device 'structure))
  :subgoals ((foreach input in (getslot device 'inputs)
    (teach particular-causal-behaviour
      (device input))))
  :template ("how" (deviceref device)
    "works given different inputs."))

(defplan particular-causal-behaviour (device)
  :subgoals ((call clearvalues)
    (foreach input window
      in input (getslot device 'inwindows)
      (call putresult (input window)))
    (interaction teaching.exchange
      (list 'suppose device inputs)
      (teach causal-bit (device inputs))))
  :template ("how" (deviceref device)
    "works given the following inputs:" inputs))

(defplan causal-bit (device inputs)
  :subgoals ((call run (device inputs)))

```

Figure B-7: Causal Behaviour

Appendix C

Discourse Plans

The following plans are used to control interactions with the user and decisions about the use of meta-comments and discourse markers. The plans are loosely based on the hierarchical discourse models of Sinclair and Coulthard and followers, and a summary of their analysis of classroom discourse is included after the plans.

C.1 Transactions

The top level definitions control the structure of 'transactions' - such as a set of exchanges on some topic or sub-topic, or an interruption on a different sub-topic.

```
(definteraction informing.transaction (teaching-goal)
  :subgoals ((interaction boundary.exchange
                  (teaching-goal 'open))
              (interaction teaching.exchanges (teaching-goal))
              (interaction boundary.exchange
                  (teaching-goal 'close))))

(definteraction interrupting.transaction (goal oldgoal)
  :subgoals ((interaction boundary.exchange (goal 'open-int))
              (interaction teaching.exchanges (goal))
              (interaction boundary.exchange
                  (oldgoal 'close-int))))

(definteraction sub.transaction (goal type)
  :subgoals ((interaction boundary.exchange (goal type))
              (interaction teaching.exchanges (goal))))
```

These rules are loosely based on Sinclair and Coulthard's analysis of informing transactions. They claim that they consist of an opening boundary exchange, an informing exchange (which may include teacher-elicited and pupil-elicited exchanges within it) and a closing boundary exchange. The plans summarise these as teaching exchanges and boundary exchanges, and allow two special types of informing transaction — the interruption and the sub-transaction on a more detailed sub-topic. This latter category arose from the observation that in Sinclair and Coulthard's data the type of boundary exchanges observed appeared to depend on whether it was a new topic being introduced, or just a sub-topic of the existing one.

C.2 Exchanges

Exchanges are either teaching exchanges or boundary exchanges. Teaching exchanges include questions from the student, questions from the teacher and statements from the teacher. The 'teaching exchanges' plan passes control to the content plans, which eventually call individual teaching exchanges. The type of teaching exchange selected to teach some fact depends on the constraints on the plan.

```
(definteraction teaching.exchanges (goal)
  :subgoals (goal)) ;; pass control to content plans

(definteraction boundary.exchange (teaching-goal type)
  :constraints ((eq type 'close))
  :subgoals ((interaction request-close.move
                      (teaching-goal))
              (interaction answer-close.move
                      (teaching-goal))))

(definteraction boundary.exchange (goal type)
  :subgoals ((interaction framing.move (type))
              (interaction focussing.move (goal type))))

(definteraction teaching.exchange (fact)
  :constraints ((maybe-understood (concept))
                (has-question fact))
  :subgoals ((interaction teacher-elicited.exchange (fact))))

(definteraction teaching.exchange (fact)
  :subgoals ((interaction teacher-inform.exchange (fact))))
```

```

(definteraction teacher-inform.exchange (fact)
  :subgoals ((interaction teacher-inform.move (fact))
              (interaction pupil-acknowledge.move (fact))))

(definteraction teacher-elicite.exchange (fact)
  :subgoals ((interaction teacher-ask.move (fact))
              (interaction pupil-answer.move (fact))
              (interaction teacher-response.move (fact))))

(definteraction pupil-elicite.exchange (goal)
  :subgoals ((interaction pupil-question.move (goal))
              (interaction teacher-reply.move
                (goal activegoal))))

```

Note that these plans may involve specialising a plan type (such as in the teaching exchange rules) or decomposing it. In the 'eliciting' exchanges the teacher (system) may only ask for simple facts, while the pupil (user) may ask questions requiring a complex answer. This distinction is necessary to simplify the understanding of student answers, but is a severe restriction to explanatory discourse.

The plans are based on the following descriptions of exchanges in classroom discourse :

Boundary: (Frame) (Focus)
 Teaching: I (R) (F)

Teaching exchanges may be further subdivided into types —

T-Inform: I (R)
 T-Direct: I R (F)
 T-Elicit: I R F
 P-Elicit: I R
 Pupil Inform: I F

In the plans the general categories of I (Initiation), R (Response) and F (Feedback) have been replaced with more context dependent versions. The 'pupil-inform' exchange is currently not allowed and the 'teacher-direct' exchange does not apply.

C.3 Moves

The following definitions of moves within an exchange are used. First, the boundary moves:

```
(definteraction framing.move (type)
  :subgoals ((interaction marker.act (type))))

(definteraction focussing.move (goal type)
  :constraints ((eq type 'close-int))
  :subgoals ((interaction resumption.act (goal type)))

(definteraction focussing.move (goal type)
  :subgoals ((interaction meta-comment.act (goal type))
              (interaction meta-comment-assess.act (goal type))))
```

Note the use of constraints to select different specialisations of the focussing move. Further specialisation (such as selection of appropriate markers) may take place within the act's, but the resumption of previous discourse seems conceptually different from the default focussing move at the beginning of a topic.

The 'meta-comment-assess' act may be used to generate meta-comments about the difficulty or importance of a topic, but is not exploited in the current version of the system.

```
(definteraction teacher-inform.move (fact)
  :subgoals ((interaction teacher-inform.act (fact))))

(definteraction pupil-acknowledge.move (fact)
  :subgoals ((interaction pupil-acknowledge.act (fact))))

(definteraction teacher-ask.move (fact)
  :subgoals ((interaction teacher-ask.act (fact))))

(definteraction pupil-answer.move (fact)
  :subgoals ((interaction pupil-answer.act (fact))))
```



```

(definteraction teacher-response.move (fact)
  :subgoals ((interaction teacher-assess.act (fact answer))
             (interaction teacher-evaluate.act (fact answer))
             (interaction teacher-comment.act (fact answer))))

```

Note that occasionally in these definitions there are free variables such as 'answer' above. As it is not possible to unify unbound variables with values in sub-plans in the planning formalism developed, these are stored as global variables. This is not an ideal solution.

The plans above either call a single 'act' or decompose into a number of acts. Some of the following plans involve reasoning about the discourse model to make decisions about discourse moves.

```

(definteraction pupil-question.move (goal)
  :subgoals ((interaction pupil-question.act (goal))))

```

```

(definteraction teacher-reply.move (goal oldgoal)
  :constraints ((inmiddleof goal))
  :subgoals ((interaction avoid-question.exchange
                        (goal 'inmiddleof))))

```

```

(definteraction teacher-reply.move (goal oldgoal)
  :constraints ((abouttodo goal))
  :subgoals ((interaction avoid-question.exchange
                        (goal 'abouttodo))))

```

```

(definteraction teacher-reply.move (goal oldgoal)
  :subgoals ((interaction interrupting.transaction
                        (goal oldgoal))))

```

```

(definteraction avoid-question.exchange (goal type)
  :subgoals ((interaction bad-question.move (goal type))
             (interaction pupil-reply.move (goal type))
             (interaction bad-question-feedback.move
                        (goal type))))

```

```

(definteraction bad-question.move (goal type)
  :subgoals ((interaction bad-question.act (goal type))))

```

```

(definteraction pupil-reply.move (goal type)
  :subgoals ((interaction pupil-reply.act (goal type))))

```

```

(definteraction bad-question-feedback.move (goal type)
  :constraints ((eq reply nil))
  :subgoals ((interaction accept.act)))

(definteraction bad-question-feedback.move (goal type)
  :subgoals ((interaction interrupting.transaction (goal))))

```

This set of plans controls responses to user's questions. In certain cases (if the question is about to be, or is already being answered) the system will generally negotiate whether it should still be answered. Depending on the user's reply¹ the system will either answer the question or just acknowledge the user's reply and continue. These plans move away from Sinclair and Coulthard's hierarchical structure, as moves may consist of exchanges or even transactions. Intuitively, an answering move, for example, may consist of a whole discussion on some topic.

Sinclair and Coulthard define their set of moves as²:

```

framing: marker

focusing: (marker) (starter) meta-statement/conclusion (comment)

opening: (marker) (starter) directive/elicitation/informative/check
(prompt/clue) (nomination - cue, bid)

answering: (acknowledge) reply/react/acknowledge (comment)

followup: (accept) (evaluate) (comment)

```

The first set of plans above are based quite closely on these, though not including all the optional elements. However, the later plans introduce further complexity — not all questions are answered and answers may involve more than a single move.

¹Note that in the evaluated version the system would not always negotiate this — It might just avoid answering the question.

²Brackets define optional elements.

C.4 Acts

The set of linguistic acts used is similar to, but not identical to Sinclair and Coulthard's set. They are currently defined as arbitrary lisp functions which select appropriate templates and print them out. Some involve more complex reasoning, such as finding which goal the system was 'in the middle of' doing for accessing the template for the resumption act. There are about 20 such acts defined.

Two acts involve calling higher level types of discourse goal. The first is the 'comment' act, which may involve a complex transaction on some topic. The second is the 'resumption' act which is modified so if a question is asked at the end of an explanation, then the close of the topic is re-negotiated. These definitions are given below:

```
(definteraction teacher-comment.act (fact answer)
  :constraints ((response fact answer)) ;; if there's a
                                          ;; remediation strategy
  :subgoals ((interaction interrupting.transaction ;; use it
              ((response fact answer))))))

(definteraction resumption.act (type goal)
  :constraints ((null goalstack))
  :subgoals ((interaction boundary.exchange (goal 'close))))
```

Appendix D

Remediation Plans

The two remediation plans in figure 4-1 show how strategies can be defined for dealing with misconceptions. If the user is asked a question and gets the answer wrong, simply correcting the user is unlikely to have a lasting affect. However, if the system can show why it was wrong, or give a hint towards the right answer, then ask again, then this is more likely to lead to lasting understanding. In general, different strategies could be defined for the different sorts of errors. The two here deal with errors in giving the input-output behaviour of a device.

These strategies have a similar form. Both have the constraint that you shouldn't use a strategy that has been used previously in the explanation. This is a simple way of avoiding 'looping' of remediation strategies if a user consistently gets something wrong. Both present some new information (from which the answer might be deduced) then ask the question again. They could be rewritten in terms of a more general strategy which took a content plan as its argument. However, the formalism will obviously support a wide range of other strategies.

```

(defplan event-reply
  (answer response device inputs output outref inrefs)
  :constraints ((understands
    (list (getslot device 'analogy))
    'what-it-does)
    (not (includes goaltree
      (plan 'teach 'similarity
        (list device
          (getslot device 'analogy))))))
  :subgoals ((teach similarity (device
    (getslot device 'analogy)))
    (interaction marker.act ('conclusion))
    (interaction teacher-elicit.exchange
      ((list describe-event device
        inputs output outref inrefs)))))

(defplan event-reply
  (answer response device inputs output outref inrefs)
  :constraints ((getslot device 'structure)
    (not (includes goaltree
      (plan 'teach
        'particular-causal-behaviour
        (list device inputs)))))
  :subgoals ((teach particular-causal-behaviour (device inputs))
    (interaction marker.act ('conclusion))
    (interaction teacher-elicit.exchange
      ((list describe-event device
        inputs output outref inrefs)))))

```

Figure D-1: Remediation Plans

Appendix E

Device Models

The following structures define the device models used in the circuit explanations. It is a representation which could be improved greatly by having a richer hierarchy of device types. However, it serves to illustrate the basic approach. The top level definitions of the devices used in the evaluation will be followed by the circuit digrams generated by the system.

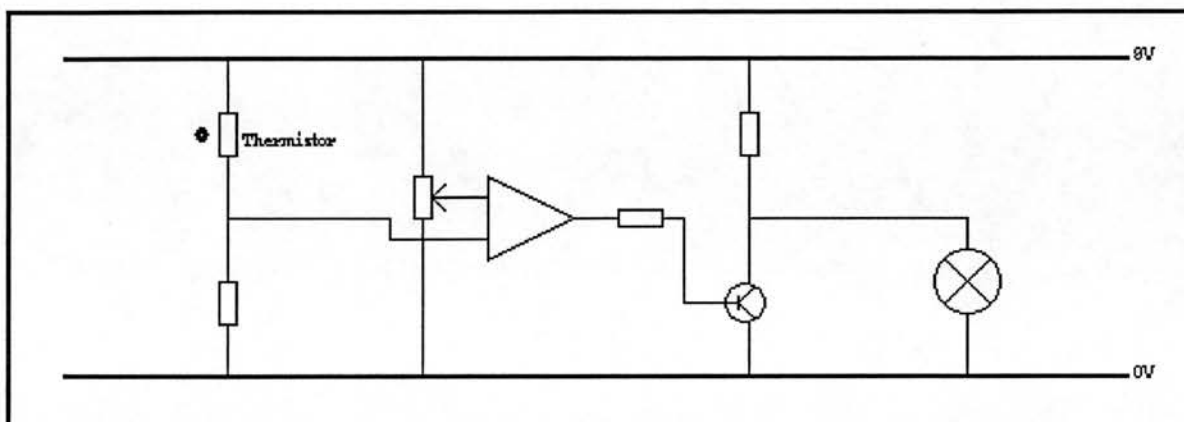
E.1 Complete Circuits

The first of these is defined out of a number of simpler circuits.

```
(defmodel heat-warning
  :name "low heat warning circuit"
  :inputs '(((heat-intensity low)) ((heat-intensity high)))
  :runfn (function run-heat-warning)
  :device-type "warning device"
  :structure '(lambda (heat)
                (lamp (inverter (comparator
                                   (heat-detector heat))))))
  :outval 'brightness
  :size '(300 . 100)
  :sort "circuit")
```

The 'size' is used in the graphics — to determine the 'active-region' which should be sensitive to mouse clicks for this device when asking questions from the diagram. The 'outval' is used to describe what the output is, while the 'sort' is used to distinguish complete circuits, circuit sections and components.

The circuit diagram of the 'low heat warning circuit' is given below:



The next five circuits are the sections of the above circuit, plus the 'light-detector' which is used in many example explanations.

```
(defmodel light-detector
  :name "light detector circuit"
  :device-type "input transducer"
  :circuit-type potential-divider
  :analogy heat-unit
  :inputs (((light-intensity high)) ((light-intensity low)))
  :runfn (function run-light-detector)
  :structure '(lambda (light) (potential-divider
                                (ldr light)
                                (resistor)))

  :outval 'voltage
  :size '(100 . 50)
  :sort "circuit section")
```

Note that the 'input transducer' is not defined as a separate model - it is therefore impossible to ask questions about it. However, the potential divider circuit type has a model definition so can be explained.

```
(defmodel heat-detector
  :name "heat detector circuit"
  :device-type "input transducer"
  :circuit-type potential-divider
  :analogy light-unit
  :inputs (((heat-intensity high)) ((heat-intensity low)))
  :runfn (function run-heat-detector)
```

```

:structure '(lambda (heat) (potential-divider
                             (thermistor heat)
                             (resistor)))

:outval 'voltage
:size '(100 . 50)
:sort "circuit section")

(defmodel inverter
  :name "inverter"
  :circuit-type potential-divider
  :inputs (((voltage high)) ((voltage low)))
  :runfn (function run-heat-detector)
  :structure '(lambda (light) (potential-divider
                              (resistor)
                              (transistor voltage)))

  :outval 'voltage
  :size '(100 . 50)
  :drawfn (function drawinverter)
  :sort "circuit section")

(defmodel comparator
  :name "comparator"
  :inputs '(((voltage highish)) ((voltage lowish)))
  :runfn (function run-comparator)
  :structure '(lambda (voltage)
                (opamp (voltage-divider (setting medium))
                       voltage))
  :drawfn (function draw-comparator)
  :outval 'voltage
  :size '(50 . 120)
  :sort "circuit section")

(defmodel lamp
  :name "lamp"
  :inputs (((voltage high)) ((voltage low)))
  :runfn (function run-lamp)
  :outval "brightness"
  :outpos '(20 . 0)
  :drawfn (function draw-lamp)
  :inposns '((-100 . 0))
  :size '(100 . 150)
  :sort "component")

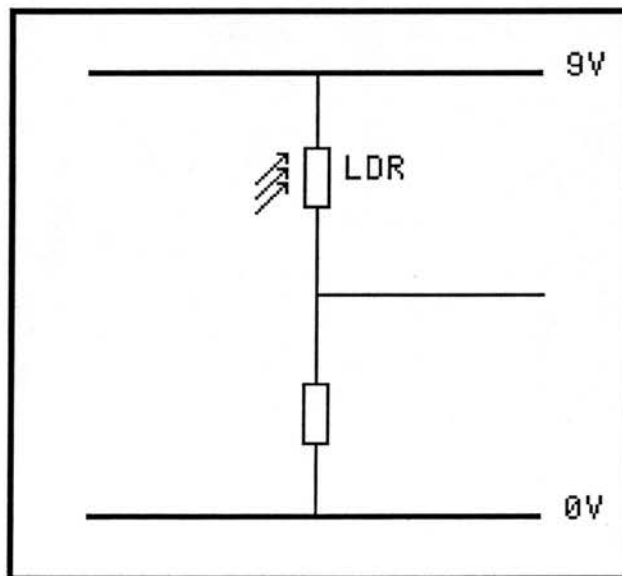
```

Note here the use of 'outpos' and 'inposns'. These are used in deciding where to position input and output meters. They need only be defined for the most

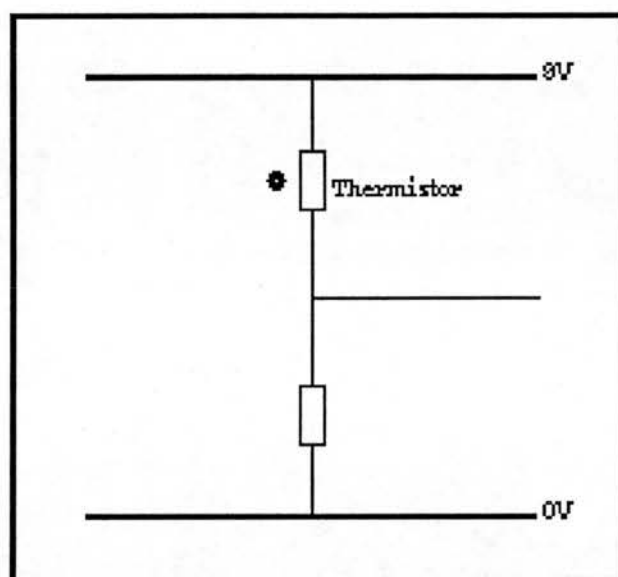
detailed components, as the higher level systems will automatically share the meters of the components.

In general the circuit digrams can be worked out (and drawn) from the structural description of the device and diagrams of components and circuit types. For example, the diagram (drawfn) for the potential divider circuit draws the basic structure with space for resistors, while the diagrams for components fill out the details. However, in some cases extra detail should be added to complete the diagram, and this is what the 'drawfn' of high level circuits may include. The following are the diagrams generated for the light detector, heat detector and inverter, all with the basic 'potential divider' structure:

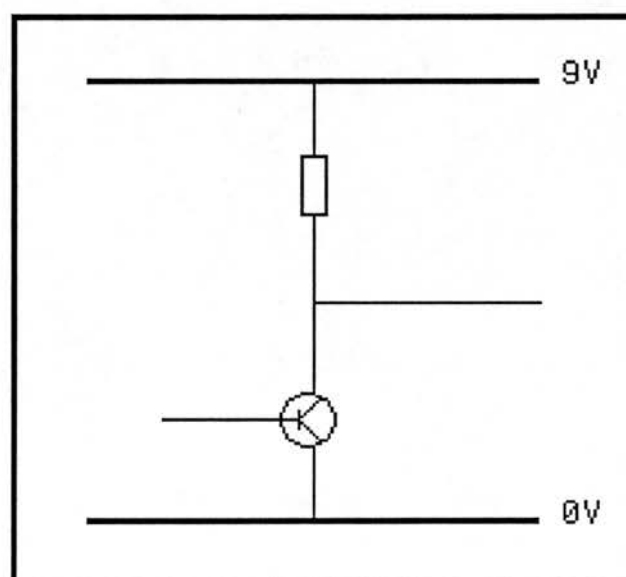
Light Detector



Heat Detector



Inverter



E.2 Components and Circuit Types

The remaining structures define the potential divider circuit and the components used in the system.

```
(defmodel potential-divider
  :name "potential divider circuit"
  :inputs '(((resistance low)(resistance high))
            ((resistance high)(resistance low)))
  :runfn (function run-pd)
  :drawfn (function draw-pd)
  :inposns '((0 . 40) (0 . -40))
  :outval 'voltage
  :outpos '(70 . 10)
  :size '(20 . 150)
  :sort "type of circuit"
  :outwindow "in the middle"
  :inwindows '("near 9V rail" "near 0V rail"))
```

Note that the general potential divider concept can be described (to some degree) by using the input and output window/meter labels in the model, rather than the labels in the particular potential divider instance.

```
(defmodel resistor
  :name "fixed resistor"
  :inputs '(nil)
  :runfn (function (lambda () '(resistance medium)))
  :drawfn (function draw-resistor)
  :outval 'resistance
  :outpos '(20 . 0)
  :size '(8 . 20)
  :sort "component")

(defmodel ldr
  :name "light dependent resistor"
  :inputs '(((light-intensity high)) ((light-intensity low)))
  :runfn (function run-ldr)
  :drawfn (function draw-ldr)
  :analogy thermistor
  :outval 'resistance
  :outpos '(20 . 0)
  :inposns '((-40 . 0))
```

```

:size '(8 . 20)
:sort "component")

(defmodel thermistor
  :name "thermistor"
  :inputs '(((heat-intensity high)) ((heat-intensity low)))
  :runfn (function run-thermistor)
  :analogy ldr
  :drawfn (function draw-thermistor)
  :outval 'resistance
  :outpos '(20 . 0)
  :inposns '((-40 . 0))
  :size '(8 . 20)
  :sort "component")

(defmodel transistor
  :name "transistor"
  :inputs '(((voltage high)) ((voltage low)))
  :runfn (function run-transistor)
  :drawfn (function draw-transistor)
  :outval 'resistance
  :outpos '(20 . 0)
  :inposns '((-40 . 0))
  :size '(15 . 15)
  :sort "component")

```

The last set of models would obviously be more efficiently defined if they were all instances of some generic resistor model, and inherited its values. This would also allow more type of explanation.

```

(defmodel opamp
  :name "OPAMP"
  :inputs '(((voltage highish)(voltage medium))
            ((voltage lowish)(voltage medium)))
  :runfn (function run-opamp)
  :drawfn (function draw-opamp)
  :outval 'voltage
  :outpos '(50 . 10)
  :inposns '((-30 . 10)(-25 . -40))
  :size '(25 . 25)
  :sort "component")

```


Appendix F

The Evaluation Questionnaire

The following questionnaire was given to subjects after using the EDGE electronics explainer for about half an hour :

Try to answer the following questions - feel free to add any related comments.

Explanation Content

How would you assess the difficulty of the explanation :

Easy ————— About right ————— Too Hard

Did you find the explanation made sense and followed on naturally?

Coherent ————— Incoherent.

Were there any things you think it should have said but didn't, or times when it told things you didn't want to know. If so, what were the main problems?

Dialogue and Interaction

Did the dialogue seem natural and easy to follow?

Natural ————— Unnatural

Did it deal with questions (interruptions) :

Well ————— Badly

Did you find the system paused when you wanted it to. If not what were the problems or what style of information presentation would you prefer?

Were there times when you wanted to ask or say something but couldn't. If so, what or when?

English and graphics

Was the use of pronouns : Good—————Bad

Was the use of meta comments and clue words (eg “We’re about to explain..”, “We were explaining”, Now, OK, Anyway) : Good—————Bad

In other respects, was the english acceptable and reasonably natural sounding. If not, what were the main problems?

Was the graphics acceptably coordinated with the English text?

What’s important

How important would you rate the following features in a tutorial explanation :

Being able to interrupt and ask questions :

Unimportant—————Important

Coordinated(ish) text and graphics in explanation :

Unimportant—————Important

Being asked questions as the explanation proceeds :

Unimportant—————Important

What you now know

Now indicate what you think you now know. This may well be things you already knew before the session. For each thing, indicate whether you think you know it (Yes), you might know know it but arent confident (?) or think you dont know it (No) :

(This followed by a list of the concepts known by the system and taught in the explanation)

Appendix G

Example Dialogues with Real Users

The following extracts are taken from the scripts of people using the system. They are chosen to illustrate problems as well as features of the system! Each extract is followed by comments.

G.1 User P: Managing Mixed Initiative Interaction

This first example illustrates some of the problems when the user's questions and the systems explanation plan overlap. Should it still say what it was going to say? What criteria can be used to modify and revise plans based on questions already answered?

- 1 OK, we're going to be explaining how the heat detector works.
- 2 Do you know what the heat detector circuit is like?
- 3 — No
- 4 A heat detector circuit is a sort of potential divider circuit.
- 5 — What does a heat detector circuit do?
- 6 I'll be getting on to that in a minute.
- 7 Do you know what the potential divider circuit does for different inputs?
- 8 — No
- 9 Oh well, let's go through what the potential divider circuit does.
- 10 When it has one low (fully conducting) input resistance (near 9V rail) and one high (not conducting) input resistance (near 0V rail) then the output voltage (in the middle) is high (9V).
- 11 When it has one high (not conducting) input resistance (near 9V rail) and one low (not conducting) input resistance (near 0V rail) then the output voltage (in the middle) is high (9V).
- 12 The heat detector's circuit is to provide an output voltage which

depends on the input heat intensity.

13 This component here is a thermistor.

14 — **How does the thermistor work.**

15 Well, a thermistor is a bit complicated to explain in detail so I'll just tell you what it does.

16 When the input heat intensity (H1) of the thermistor is high (hot) the output resistance (R1) will be quite low.

17 When it is low (cold) the output resistance (R1) will be quite high.

18 Anyway, we were in the middle of describing the component: thermistor.

19 Let's go through what the thermistor does.

20 What's the output resistance (R1) of the thermistor going to be when the input heat intensity is high (hot).

21 — **Quite low**

22 Yes, splendid, it's quite low.

23 What's it going to be when the input heat intensity is low (cold).

24 — **Quite high**

25 Yes, great, it's quite high.

26 Now, we're going to be explaining the behaviour of the heat detector circuit

It is clear that this example does not deal especially well with the problem. In line 6 a question is deferred because it is already in the plan - yet maybe the student knows better than the system when a particular bit of knowledge should be explained. It would probably have been better to negotiate the deferral. In line 19 and following the system essentially goes over the same content again, after an interruption. While the system will avoid re-explaining a goal which was directly asked by the user, it will re-explain subgoals - yet this is not always appropriate.

It may prove impossible to provide completely subject independent rules of this kind, though several improvements could be made to the existing system. One improvement would result from allowing partially ordered plans. Then goals in the plan could be 'brought forward' when the user asks a question.

G.2 User M: Followup Questions and Remediation

The next example illustrates one way that the system responds to incorrect answers to questions, and how questions may be asked at the end of an explanation. In this session the 'heat warning' circuit is being explained. The explanation has progressed by the system explaining the behaviour of the circuit at a high level of detail (in terms of circuit sections) and asking the user questions. When the

user fails to answer correctly the system shifts down a level, explaining at a lower level of detail so the user can follow through why the answer was incorrect. The conclusion of the explanation is given below.

- 1 What's the output voltage (V7) of the inverter going to be when the input voltage (V5) is low (0V).
- 2 — **Quite high (towards 9V)**
- 3 Nope, you lose,
- 4 OK, maybe we better go through what the inverter does when it has a low input voltage.
- 5 Suppose the input voltage is low (0V).
- 6 When the input voltage (V5) of the transistor is low (0V) the output resistance (R4) will be high (not conducting).
- 7 When the potential divider circuit has one average (a few KOhm) input resistance (R3) and one high (not conducting) input resistance (R4) then the output voltage (V7) is high (9V).
- 8 Anyway, we were in the middle of explaining how the low heat warning circuit works if it has the following inputs: heat intensity high.
- 9 When the input voltage (V7) of the lamp is high (9V) the output brightness (B1) will be high (lamp on).
- 10 So,
- 11 When the input heat intensity (H1) of the low heat warning circuit is low (cold) the output brightness (B1) will be high (lamp on). When it is high (hot) the output brightness (B1) will be low (lamp off).
- 12 Is that enough about how the low heat warning circuit works.
- 13 — **What does the comparator do**
- 14 OK, When the input voltage (V2) of the comparator is quite high (towards 9V) the output voltage (V5) will be low (0V).
- 15 When it is quite low (towards 0V) the output voltage (V5) will be high (9V).
- 16 Anyway, is that enough about that?
- 17 — **How does the comparator work.**
- 18 OK, suppose the input voltage is quite high (towards 9V).
-
- 19 Anyway, is that enough about that.
- 20 — **OK**

This dialogue shows how errors may be explained and problems dealt with in followup questions. There are no major problems with the interaction, yet the explanation is still weak compared with human explanations. This seems largely due to the simplicity of the underlying models explaining everything in terms of inputs and outputs.

G.3 User D: Managing Interruptions

This example shows a two remediation strategies being used, and problems when the user asks consecutive questions! The first remediation strategy is clear and apparently helpful (lines 4-14). The second is based on over simple models and is not so good (lines 20-21). In line 23 the user interprets the question 'what's a comparator' differently to the system, so needs so ask another question (what it does). This again is not really answered adequately due to the over simple models - a more appropriate answer might be 'A comparator is a device which compares its two inputs and gives either a high or low voltage depending on which is greater. The user accidentally asks the question again, but the system correctly recognises this and avoids it. However, the nested interruptions result in an incorrect 'resumption' of two anyways.

- 1 Suppose the input heat intensity is high (hot).
- 2 What's the output voltage (V2) of the heat detector circuit going to be when the input heat intensity is high (hot).
- 3 — **Low (0V).**
- 4 Wrong, not quite, OK,
- 5 A heat detector circuit is much like a light detector circuit only:
- 6 - Its output voltage depends on the input light intensity.
- 7 - It has a thermistor instead of a light dependent resistor.
- 8 This component here is a thermistor.
- 9 Let's go through what a thermistor does.
- 10 When the input heat intensity (H1) of the thermistor is high (hot) the output resistance (R1) will be quite low.
- 11 When it is low (cold) the output resistance (R1) will be quite high.
- 12 So, what's the output voltage (V2) of the heat detector circuit going to be when the input heat intensity is high (hot).
- 13 — **Quite high (towards 9V).**
- 14 Precisely, very good, it's quite high (towards 9V).
- 15 Anyway, we were in the middle of explaining how the low heat warning device works if it has the following inputs: heat intensity high.
- 16 What's the output voltage (V5) of the comparator going to be when the input voltage (V2) is quite high (towards 9V).
- 17 — **Quite high**
- 18 Wrong, you lose,
- 19 OK, maybe we better go through what a comparator does when it has a highish input voltage.
- 20 Suppose the input voltage is quite high.
- 21 When the operational amplifier has one medium (about 4.5V) input voltage (v3) and one quite high input voltage (V2) then the

output voltage (V5) is low (0V).

22 Anyway, we were in the middle of explaining how the low heat warning device works if it has the following inputs: heat intensity.

23 — **What's a comparator.**

24 OK, this circuit section here is a comparator.

25 — **What does a comparator do.**

26 OK, when the input voltage (V2) of the comparator is quite high (towards 9V) the output voltage (V5) will be low (0V).

27 — **What does a comparator do.**

28 I'm in the middle of saying that. Do you want it answered right now?

29 — **No**

30 When it is quite low (towards 0V) the output voltage (V5) will be high (9V).

31 Anyway, we were in the middle of explaining how the low heat warning device works if it has the following inputs: heat intensity high.

32 Anyway, we were in the middle of explaining how the low heat warning device works if it has the following inputs: heat intensity high.

33 What's the output voltage of the inverter

G.4 User J: Repeated Explanations

This example illustrates what happens when the user asks for a circuit to be explained for a second time. The first time the circuit was explained relatively few questions were asked by the system, but several by the user. The system has therefore built up a set of assumptions about what the user may know.

The example also illustrates various points about use of pronouns and includes an interrupted question sequence, discussed below.

1 OK, we're going to be explaining how the light detector circuit works.

2 It is a sort of potential divider circuit.

3 Let's go through what the potential divider circuit does.

4 What's the output voltage (in the middle of) the potential divider circuit going to be when there is a low (fully conducting) input resistance (near 9V rail) and a high (not conducting) input resistance (near 0V rail)?

5 — **Quite high (towards 9V).**

6 No, I'm afraid not, it's high (9V).

7 When it has one high (not conducting) input resistance (near 9V rail) and one low (fully conducting) input resistance (near 0V rail)

then it is low (0V).

8 Can you tell me what the output voltage of the light detector circuit depends on?

9 — **Light intensity.**

10 Yes, splendid, it's light intensity.

11 This component here is a light dependent resistor.

12 Let's go through what the light dependent resistor does.

13 When the input light intensity (L1) of the light dependent resistor is high (light) the output resistance (R1) will be quite low.

14 What's it going to be when the input light intensity (L1) is low (dark)?

15 — **Quite high**

16 Yes, splendid, it's quite high.

17 OK, we're going to explain the behaviour of the light detector circuit.

18 Suppose the input light intensity is high (light).

19 When the input light intensity (L1) of the light dependent resistor is high (light) the output resistance (R1) will be quite low.

20 What's the output voltage (V1) of the potential divider circuit going to be when there is a quite high input resistance (R1) and a average (a few KOhm) input resistance (R2)?

21 — **Quite high**

22 Correct, great, it's quite high.

23 Suppose the input light intensity is low (dark).

24 When the input light intensity (L1) of the light dependent resistor is low (dark) the output resistance (R1) will be quite high.

25 What's the output voltage (V1) of the potential divider circuit going to be when there is a quite low input resistance (R1) and a average (a few KOhm) input resistance (R2)?

26 — **Quite low**

27 Precisely, very good, it's quite low.

28 So,

29 What's it going to be when the input light intensity (L1) is high (light).

30 — **Quite low (towards 0V)**

31 No, you lose, it's quite high (towards 0V).

32 What's it going to be when the input light intensity (L1) is low (dark)

33 — **What's a resistor?**

34 OK, this component here is a resistor.

35 Anyway, what's the output voltage (V1) of the light detector circuit going to be when the input light intensity (L1) is low (dark)?

36 — **Quite low (towards 0V)**

37 Precisely, excellent, it's quite low.

38 Is that enough about how the light detector circuit works?

39 — OK.

Note that in lines 11 to 13 pronominalisation is not used correctly. This is actually because it is not included in meta-comments currently, though should be! In line 29 the error is more fundamental - a pronoun is used because 'V1' was in the previous utterance, yet it is clearly inappropriate for two possible reasons. First, it is part of a new discourse segment, so according to Grosz and Sidner cannot refer back across segments. Second, it refers to the same voltage, but this is now viewed as 'the output of the light detector' and not the output of the potential divider. In line 35 the system uses the full expression after the interruption - this seems correct, yet Grosz and Sidner's model would allow a pronoun to be used here. A reasonable rule to improve pronoun use (conservatively) might be 'Use a pronoun for the principle focus if it is mentioned in the last sentence UNLESS it is in a separate discourse segment'.

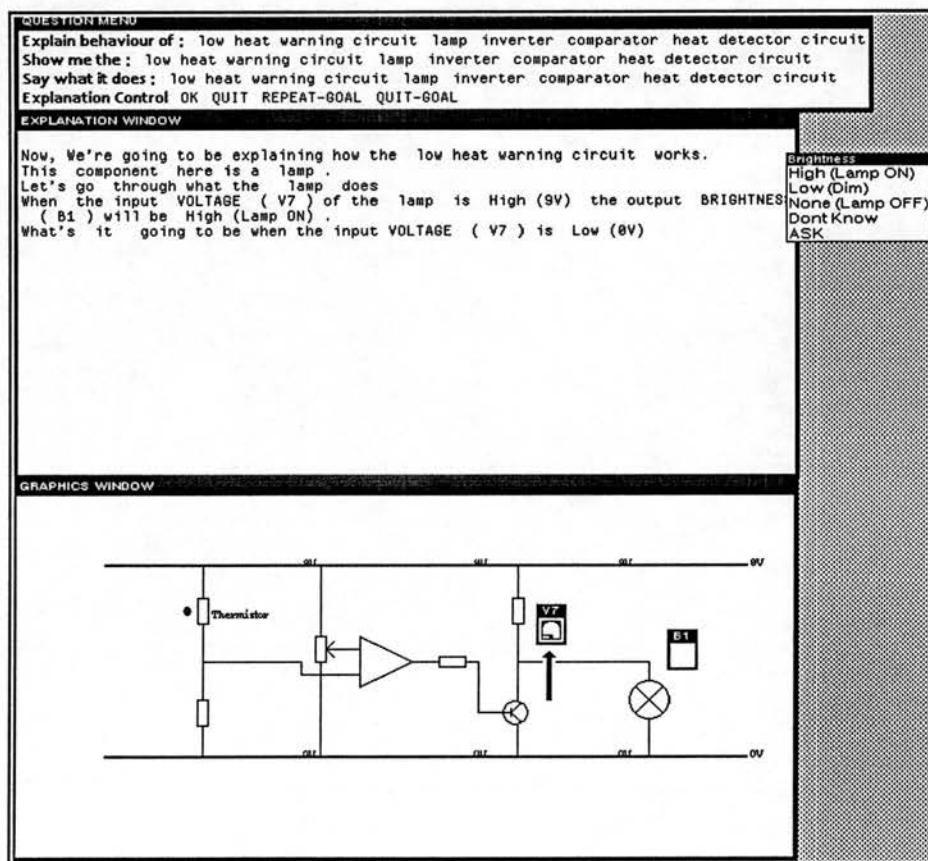
Lines 32-35 illustrate how the system deals with questions following questions - the strategy is to treat it as an interruption, but to re-ask the question after the interruption.

At the end of the session the system will have concluded that the user knows what a potential divider circuit and a light dependent resistor do. It will also have adjusted its assessment of the user's level of expertise. For example, when the user asks 'what's a resistor' this will cause the level to be decreased because the user is asking an 'easy' question. Neither of these assessments is very accurate - the questions asked by the user depend on his interactional style as well as his knowledge, and many users can give correct answers to questions based on studying the explanation and guesswork, without understanding or recalling. However, the accuracy is not crucial because of the possibilities of interaction - if it is better than random then it is worth using.

Appendix H

Example Displays

The figures below are example screen displays when the low heat warning' device is explained. The first shows a question being asked in a component description.



This next example again shows a question asked by the system, but this time as part of the causal explanation of the whole circuit.

QUESTION MENU
Explain behaviour of: low heat warning circuit lamp inverter comparator heat detector circuit
Show me the: low heat warning circuit lamp inverter comparator heat detector circuit
Say what it does: low heat warning circuit lamp inverter comparator heat detector circuit
Explanation Control OK QUIT REPEAT-GOAL QUIT-GOAL

EXPLANATION WINDOW
 This circuit section here is a comparator .
 Let's go through what the comparator does
 When the input VOLTAGE (V2) of the comparator is Quite High (Towards 9V)
 the output VOLTAGE (V5) will be Low (0V) .
 When it is Quite Low (Towards 0V) the output VOLTAGE (V5) will be
 High (9V) .
 This circuit section here is a heat detector circuit .
 Do you know what the heat detector circuit does for different inputs
 OK, Let's go through what the heat detector circuit does
 When the input HEAT-INTENSITY (H1) of the heat detector circuit is
 High (Hot) the output VOLTAGE (V2) will be Quite High (Towards 9V) .
 When it is Low (Cold) the output VOLTAGE (V2) will be
 Quite Low (Towards 0V) .
 RIGHT, We should explain the behaviour of the low heat warning circuit

 Suppose the input HEAT-INTENSITY is Low (Cold) .
 What's the output VOLTAGE (V2) of the heat detector circuit
 going to be when the input HEAT-INTENSITY (H1) is Low (Cold)

GRAPHICS WINDOW

Voltages
 High (9V)
 Quite High (Towards 9V)
 Medium (About 4.5V)
 Quite Low (Towards 0V)
 Low (0V)
 Don't Know
 ASK

Finally, this example shows how the user can ask questions from the diagram. A question menu is displayed if the user points at a particular component — in this case a transistor.

